ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

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

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUI

Ngành: Công nghệ thông tin

HÀ NỘI - 2005

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

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

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: ThS. Phan Xuân Hiếu

HÀ NỘI - 2005

Lời cảm ơn

Trước tiên, em muốn gửi lời cảm ơn sâu sắc nhất đến thầy giáo, TS. Hà Quang Thụy và ThS. Phan Xuân Hiếu, 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" như ThS.Nguyễn Trí Thành, ThS. Tào Thị Thu Phượng, CN. Vũ Bội Hằng, CN. Nguyễn Thị Hương Giang ... 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 em trai, những người luôn kịp thời động viên và giúp đỡ em vượt qua những khó khăn trong cuộc sống.

Sinh Viên

Nguyễn Cẩm Tú

Tóm tắt

Nhận biết các loại thực thể là một bước cơ bản trong trích chọn thông tin từ văn bản và xử lý ngôn ngữ tự nhiên. Nó được ứng dụng nhiều trong dịch tự động, tóm tắt văn bản, hiểu ngôn ngữ tự nhiên , nhận biết tên thực thể trong sinh/y học và đặc biệt ứng dụng trong việc tích hợp tự động các đối tượng, thực thể từ môi trường Web vào các ontology ngữ nghĩa và các cơ sở tri thức.

Trong khóa luận này, em trình bày một giải pháp nhận biết loại thực thể cho các văn bản tiếng Việt trên môi trường Web. Sau khi xem xét các hướng tiếp cận khác nhau, em chọn phương pháp tiếp cận học máy bằng cách xây dựng một hệ thống nhận biết loại thực thể dựa trên mô hình Conditional Random Fields (CRF- Laferty, 2001) . Điểm mạnh của CRF là nó có khả năng xử lý dữ liệu có tính chất chuỗi, có thể tích hợp hàng trăm nghìn thậm chí hàng triệu đặc điểm từ dữ liệu hết sức đa dạng nhằm hỗ trợ cho quá trình phân lớp. Thực nghiệm trên các văn bản tiếng Việt cho thấy qui trình phân lớp đạt được kết quả rất khả quan.

Mục lục

Bảng từ viết tắt

Từ hoặc cụm từ	Viết tắt
Conditional Random Field	CRF
Mô hình Markov ẩn	HMM
Mô hình Markov cực đại hóa entropy	MEMM

Mở đầu

Tim Benner Lee, cha đẻ của World Wide Web hiện nay, đã đề cập Web ngữ nghĩa như là tương lai của World Wide Web, trong đó nó kết hợp khả năng hiểu được bởi con người và khả năng xử lý được bởi máy. Thành công của Web ngữ nghĩa phụ thuộc phần lớn vào các ontology cũng như các trang Web được chú giải theo các ontology này. Trong khi những lợi ích mà Web ngữ nghĩa đem lại là rất lớn thì việc xây dựng các ontology một cách thủ công lại hết sức khó khăn. Giải pháp cho vấn đề này là ta phải dùng các kĩ thuật trích chọn thông tin nói chung và nhận biết các loại thực thực thể nói riêng để tự động hóa một phần quá trình xây dựng các ontology. Các ontology và hệ thống nhận biết các loại thực thể khi được tích hợp vào máy tìm kiếm sẽ làm tăng độ chính xác của tìm kiếm và cho phép tìm kiếm hướng thực thể, khắc phục được một số nhược điểm cho các máy tìm kiếm dựa trên từ khóa hiện nay.

Ý thức được những lợi ích mà các bài toán trích chọn thông tin nói chung và nhận biết loại thực thể nói riêng, em đã chọn hướng nghiên cứu nhằm giải quyết bài toán nhận biết loại thực thể cho tiếng Việt làm đề tài luận văn của mình.

Luận văn được tổ chức thành 5 chương như sau:

- *Chương 1* giới thiệu về bài toán trích chọn thông tin và bài toán nhận diện các loại thực thể cùng những ứng dụng của nó.
- *Chương 2* trình bày một số hướng tiếp cận nhằm giải quyết bài toán nhận biết loại thực thể như phương pháp thủ công, các phương pháp học máy HMM và MEMM. Các hướng tiếp cận thủ công có nhược điểm là tốn kém về mặt thời gian, công sức và không khả chuyển. Các phương pháp học máy như HMM hay MEMM tuy có thể khắc phục được nhược điểm của hướng tiếp cận thủ công nhưng lại gặp phải một số vấn đề do đặc thù của mỗi mô hình. Với HMM, ta không thể tích hợp các thuộc tính lồng nhau mặc dù những thuộc tính này rất hữu ích cho quá trình gán nhãn dữ liệu dạng chuỗi. MEMM ,trong một số trường hợp đặc biệt, gặp phải vấn đề "label bias", đó là xu hướng bỏ qua các dữ liệu quan sát khi trạng thái có ít đường đi ra.
- *Chương 3* giới thiệu định nghĩa CRF, nguyên lý cực đại hóa Entropy – một phương pháp đánh giá phân phối xác suất từ dữ liệu và là cơ sở để chọn các "hàm tiềm năng" cho các mô hình CRF, thuật toán Viterbi để gán nhãn cho dữ liệu dạng chuỗi. Bản chất "phân phối điều kiện" và "phân phối toàn cục" của CRF cho phép các mô hình này khắc phục được các nhược điểm của các mô

hình học máy khác như HMM và MEMM trong việc gán nhãn và "phân đoạn" (segmentation) các dữ liệu dạng chuỗi.

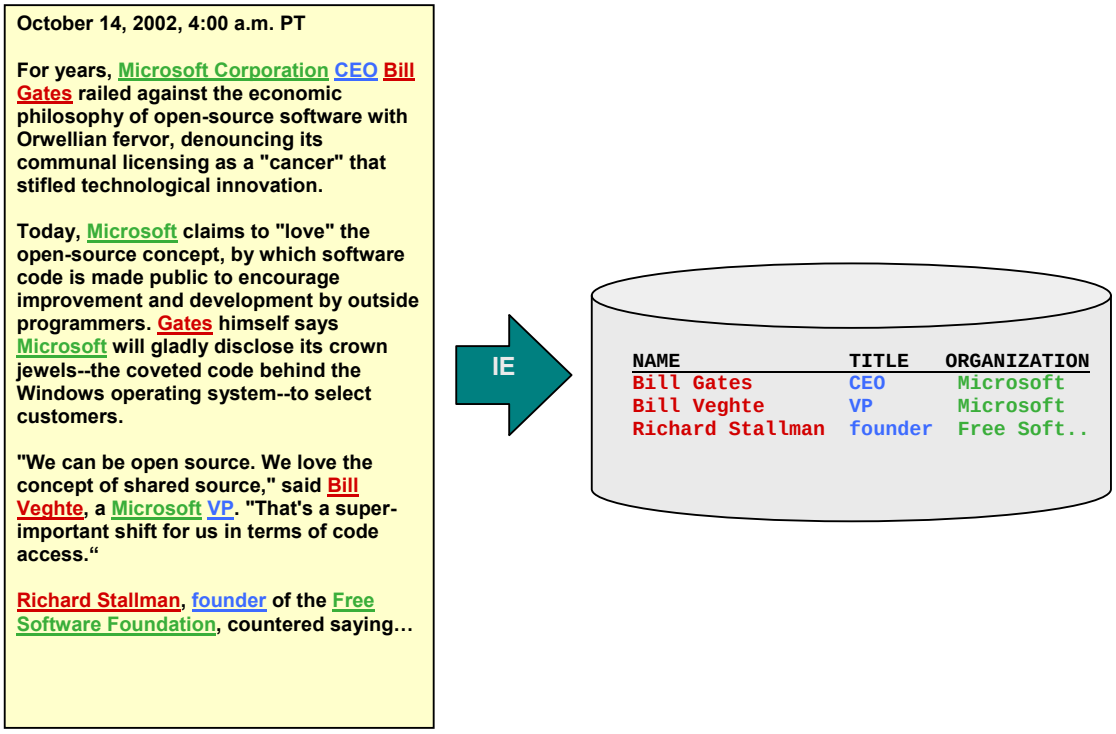
- *Chương 4* trình bày những phương pháp để ước lượng các tham số cho mô hình CRF như các thuật toán IIS, GIS, các phương pháp dựa trên vector gradient như phương pháp "gradient liên hợp", quasi-Newton, L-BFGs. Trong số các phương pháp này, phương pháp L-BFGs được đánh giá là tốt nhất và có tốc độ hội tụ nhanh nhất.
- *Chương 5* trình bày hệ thống nhận diện loại thực thể cho tiếng Việt dựa trên mô hình CRF, đề xuất các phương pháp chọn thuộc tính cho việc nhận biết các loại thực thể trong các văn bản tiếng Việt và đưa ra một số kết quả thực nghiệm.

Chương 1. Bài toán nhận diện loại thực thể

Chủ đề chính của khóa luận là áp dụng mô hình CRF cho bài toán nhận biết các loại thực thể cho tiếng Việt. Chương này sẽ giới thiệu tổng quan về trích chọn thông tin [30][31][32], chi tiết về bài toán nhận biết loại thực thể [13][15][30][31] và những ứng dụng của bài toán nhận biết loại thực thể.

1.1. Trích chọn thông tin

Không giống như việc hiểu toàn bộ văn bản, các hệ thống trích chọn thông tin chỉ cố gắng nhận biết một số dạng thông tin đáng quan tâm. Có nhiều mức độ trích chọn thông tin từ văn bản như xác định các thực thể (Element Extraction), xác định quan hệ giữa các thực thể (Relation Extraction), xác định và theo dõi các sự kiện và các kịch bản (Event and Scenario Extraction and Tracking), xác định đồng tham chiếu (Co-reference Resolution) ... Các kĩ thuật được sử dụng trong trích chọn thông tin gồm có: phân đoạn, phân lớp, kết hợp và phân cụm.



Hình 1: Một hệ thống trích chọn thông tin

Kết quả của một hệ thống trích chọn thông tin thường là các mẫu (template) chứa một số lượng xác định các trường (slots) đã được điền thông tin.

Ở mức độ trích chọn thông tin ngữ nghĩa, một mẫu là thể hiện của một sự kiện trong đó các thực thể tham gia đóng một số vai trò xác định trong sự kiện đó. Chẳng hạn như tại MUC-7 [31] (Seventh Message Understanding Conference), một mẫu kịch bản được yêu cầu là các sự kiện phóng tên lửa và rocket trong 100 bài báo của New York Times. Các hệ thống tham gia hội nghị phải điền vào mẫu này các thông tin sao cho có thể trả lời được câu hỏi về thời gian, địa điểm ... của các sự kiện phóng tên lửa, rocket được đề cập trong các bài báo.

1.2. Bài toán nhận biết các loại thực thể

Con người, thời gian, địa điểm, các con số, ... là những đối tượng cơ bản trong một văn bản dù ở bất kì ngôn ngữ nào. Mục đích chính của bài toán nhận biết các loại thực thể là xác định những đối tượng này từ đó phần nào giúp cho chúng ta trong việc hiểu văn bản.

Bài toán nhận biết các loại thực thể là bài toán đơn giản nhất trong số các bài toán trích chọn thông tin, tuy vậy nó lại là bước cơ bản nhất trước khi tính đến việc giải quyết các bài toán phức tạp hơn trong lĩnh vực này. Rõ ràng trước khi có thể xác định được các mối quan hệ giữa các thực thể ta phải xác định được đâu là các thực thể tham gia vào mối quan hệ đó.

Tuy là bài toán cơ bản nhất trong trích chọn thông tin, vẫn tồn tại một lượng lớn các trường hợp nhập nhằng làm cho việc nhận biết các loại thực thể trở nên khó khăn. Một số ví dụ cụ thể :

- "Bình Định và HAGL đều thua ở AFC Champion Ledge ".
  - Ở đây "Bình Định" phải được đánh dấu là một tổ chức (một đội bóng) thay vì là một địa danh.
  - Chữ "Bình" viết đầu câu nên thông tin viết hoa không mang nhiều ý nghĩa.
- Khi nào "Hồ Chí Minh" được sử dụng như tên người, khi nào được sử dụng như tên một địa danh?

Bài toán nhận biết loại thực thể trong các văn bản tiếng Việt còn gặp nhiều khó khăn hơn so với bài toán này trong tiếng Anh vì một số nguyên nhân như sau:

- Thiếu dữ liệu huấn luyện và các nguồn tài nguyên có thể tra cứu như WordNet trong tiếng Anh.

- ❖ Thiếu các thông tin ngữ pháp (POS) và các thông tin về cụm từ như cụm danh từ, cụm động từ ... cho tiếng Việt trong khi các thông tin này giữ vai trò rất quan trọng trong việc nhận biết loại thực thể.

Ta hãy xem xét ví dụ sau: "Cao Xumin, Chủ tịch Phòng Thương mại Xuất nhập khẩu thực phẩm của Trung Quốc, cho rằng cách xem xét của DOC khi đem so sánh giá tôm của Trung Quốc và giá tôm của Ấn Độ là vi phạm luật thương mại"

Chúng ta muốn đoạn văn bản trên được đánh dấu như sau: "**<PER>** Cao Xumin**</PER>**, Chủ tịch **<ORG>**Phòng Thương mại Xuất nhập khẩu thực phẩm **</ORG>** của **<LOC>**Trung Quốc**</LOC>**, cho rằng cách xem xét của **<ORG>**DOC**</ORG>** khi đem so sánh giá tôm của **<LOC>**Trung Quốc**</LOC>** và giá tôm của **<LOC>**Ấn Độ**</LOC>** là vi phạm luật thương mại"

Ví dụ trên đã bộc lộ một số khó khăn mà một hệ thống nhận biết các loại thực thể tiếng Việt gặp phải trong khi gán nhãn cho dữ liệu (xem phụ lục):

- ❖ Cụm từ "Phòng Thương mại Xuất nhập khẩu thực phẩm" là tên một tổ chức nhưng không phải từ nào cũng viết hoa.
- ❖ Các thông tin như "Phòng Thương mại Xuất nhập khẩu thực phẩm" là một cụm danh từ và đóng vai trò chủ ngữ trong câu rất hữu ích cho việc đoán nhận chính xác loại thực thể, tuy vậy do tiếng Việt thiếu các hệ thống tự động đoán nhận chức năng ngữ pháp và cụm từ nên việc nhận biết loại thực thể trở nên khó khăn hơn nhiều so với tiếng Anh.

1.3. Mô hình hóa bài toán nhận biết các loại thực thể

Bài toán nhận biết loại thực thể trong văn bản là tìm câu trả lời cho các câu hỏi: ai?, bao giờ?, ở đâu?, bao nhiêu? ... Đây là một trường hợp cụ thể của bài tóan gán nhãn cho dữ liệu dạng chuỗi, trong đó (trừ nhãn O) thì mỗi một nhãn gồm một tiếp đầu ngữ B_ hoặc I_ (với ý nghĩa là bắt đầu hay bên trong một tên thực thể) kết hợp với tên nhãn.

Bảng 1: Các loại thực thể

Tên nhãn	Ý nghĩa
PER	Tên người
ORG	Tên tổ chức

LOC	Tên địa danh
NUM	Số
PCT	Phần trăm
CUR	Tiền tệ
TIME	Ngày tháng, thời gian
MISC	Những loại thực thể khác ngòai 7 lọai trên
O	Không phải thực thể

Ví dụ: chuỗi các nhãn tương ứng cho cụm "Phan Văn Khải" là "B_PER I_PER I_PER"

Như vậy với 8 loại thực thể kể cả Misc, ta sẽ có tương ứng 17 nhãn (8*2+1). Về bản chất gán nhãn cho dữ liệu là chính là một trường hợp đặc biệt của phân lớp trong văn bản, ở đây các lớp chính là các nhãn cần gán cho dữ liệu.

1.4. Ý nghĩa của bài toán nhận biết các loại thực thể

Một hệ thống nhận biết các loại thực thể tốt có thể được ứng dụng trong nhiều lĩnh vực khác nhau, cụ thể nó có thể được sử dụng nhằm:

- Hỗ trợ Web ngữ nghĩa. Web ngữ nghĩa là các trang Web có thể biểu diễn dữ liệu "thông minh" , ở đây "thông minh" chỉ khả năng kết hợp, phân lớp và khả năng suy diễn trên dữ liệu đó. Sự thành công của các Web ngữ nghĩa phụ thuộc vào các ontology [] cũng như sự phát triển của các trang Web được chú giải bởi các siêu dữ liệu tuân theo các ontology này. Mặc dù các lợi ích mà các ontology đem lại là rất lớn nhưng việc xây dựng chúng một cách tự động lại hết sức khó khăn. Vì lý do này, các công cụ trích chọn thông tin tự động từ các trang Web để "làm đầy " các ontology như hệ thống nhận biết các loại thực thể là hết sức cần thiết.
- Xây dựng các máy tìm kiếm hướng thực thể. Người dùng có thể tìm thấy các trang Web nói về "Clinton" là một địa danh ở Bắc Carolina một cách nhanh chóng mà không phải duyệt qua hàng trăm trang Web nói về tổng thống Bill Clinton.

- ❖ Nhận biết các loại thực thể có thể được xem như là bước tiền xử lý làm đơn giản hóa các bài toán như dịch máy, tóm tắt văn bản ...
- ❖ Như đã được đề cập trên đây, một hệ thống nhận biết các loại thực thể có thể đóng vai trò là một thành phần cơ bản cho các bài toán trích chọn thông tin phức tạp hơn.
- ❖ Trước khi đọc một tài liệu, người dùng có thể đọc lướt qua các tên người, tên địa danh, tên công ty được đề cập đến trong đó.
- ❖ Tự động đánh chỉ số cho các sách. Trong các sách, phần lớn các chỉ mục là các loại thực thể.

Hệ thống nhận diện loại thực thể cho tiếng Việt sẽ làm tiền đề cho việc giải quyết các bài toán về trích chọn thông tin từ các tài liệu tiếng Việt cũng như hỗ trợ cho việc xử lý ngôn ngữ tiếng Việt. Áp dụng hệ thống để xây dựng một ontology về các thực thể trong tiếng Việt sẽ đặt nền móng cho một thế hệ Web mới - " Web ngữ nghĩa tiếng Việt".

Chương 2. Các hướng tiếp cận giải quyết bài toán nhận biết các loại thực thể

Có nhiều phương pháp tiếp cận khác nhau để giải quyết bài toán nhận diện các loại thực thể, chương này sẽ giới thiệu một số hướng tiếp cận như vậy cùng với những ưu nhược điểm của chúng từ đó lý giải tại sao chúng em lại chọn phương pháp dựa trên CRF để xây dựng hệ thống nhận diện loại thực thể cho tiếng Việt.

2.1. Hướng tiếp cận thủ công

Tiêu biểu cho hướng tiếp cận thủ công là hệ thống nhận biết loại thực thể Proteous của đại học New York tham gia MUC-6. Hệ thống được viết bằng Lisp và được hỗ trợ bởi một số lượng lớn các luật. Dưới đây là một số ví dụ về các luật được sử dụng bởi Proteous cùng với các trường hợp ngoại lệ của chúng:

- Title Capitalized_Word => Title Person Name
  - Đúng : Mr. Johns, Gen. Schwarzkopf
  - Ngoại lệ: Mrs. Field's Cookies (một công ty)
- Month_name number_less_than_32 => Date
  - Đúng: February 28, July 15
  - Ngoại lệ: Long March 3 (tên một tên lửa của Trung Quốc).

Trên thực tế, mỗi luật trên đều chứa một số lượng lớn các ngoại lệ. Thậm chí ngay cả khi người thiết kế tìm cách giải quyết hết các ngoại lệ mà họ nghĩ đến thì vẫn tồn tại những trường hợp chỉ xuất hiện khi hệ thống được đưa vào thực nghiệm. Hơn nữa, việc xây dựng một hệ thống trích chọn dựa trên các luật là rất tốn công sức. Thông thường để xây dựng một hệ thống như vậy đòi hỏi công sức vài tháng từ một lập trình viên với nhiều kinh nghiệm về ngôn ngữ học. Thời gian này còn lớn hơn khi chúng ta muốn chuyển sang lĩnh vực khác hay sang ngôn ngữ khác.

Câu trả lời cho các giới hạn này là phải xây dựng một hệ thống bằng cách nào đó có thể "tự học", điều này sẽ giúp giảm bớt sự tham gia của các chuyên gia ngôn ngữ và làm tăng tính khả chuyển cho hệ thống. Có rất nhiều phương pháp học máy như các mô hình markov ẩn (Hidden Markov Models - HMM), các mô hình Markov cực đại hóa Entropy (Maximum Entropy Markov Models- MEMM) và mô hình Conditional Random Field (CRF)... có thể được áp dụng để giải quyết bài toán nhận biết loại thực thể. Các mô hình CRF sẽ được miêu tả chi tiết trong chương sau, ở đây

chúng ta sẽ chỉ xem xét các mô hình HMM và MEMM cùng với ưu và nhược điểm của chúng.

2.2. Các mô hình Markov ẩn (HMM)

Mô hình Markov[7][13][19] ẩn được giới thiệu và nghiên cứu vào cuối những năm 1960 và đầu những năm 1970 ,cho đến nay nó được ứng dụng nhiều trong nhận dạng tiếng nói, tin sinh học và xử lý ngôn ngữ tự nhiên.

2.2.1. Tổng quan về các mô hình HMM

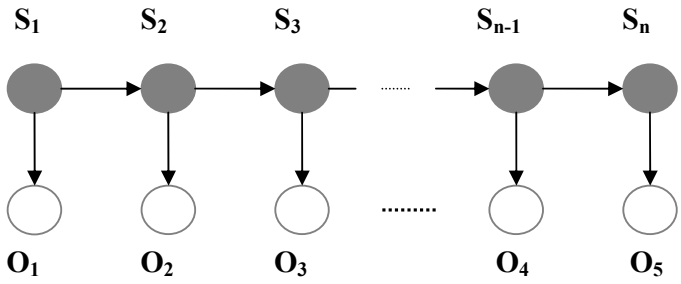
HMM là mô hình máy trạng thái hữu hạn (probabilistic finite state machine) với các tham số biểu diễn xác suất chuyển trạng thái và xác suất sinh dữ liệu quan sát tại mỗi trạng thái.

Các trạng thái trong mô hình HMM được xem là bị ẩn đi bên dưới dữ liệu quan sát sinh ra do mô hình. Quá trình sinh ra chuỗi dữ liệu quan sát trong HMM thông qua một loạt các bước chuyển trạng thái xuất phát từ một trong các trạng thái bắt đầu và dừng lại ở một trạng thái kết thúc. Tại mỗi trạng thái, một thành phần của chuỗi quan sát được sinh ra trước khi chuyển sang trạng thái tiếp theo. Trong bài toán nhận biết loại thực thể, ta có thể xem tương ứng mỗi trạng thái với một trong nhãn B_PER, B_LOC, I_PER...và dữ liệu quan sát là các từ trong câu. Mặc dù các lớp này không sinh ra các từ, nhưng mỗi lớp được gán cho một từ bất kì có thể xem như là sinh ra từ này theo một cách thức nào đó. Vì thế ta có thể tìm ra chuỗi các trạng thái (chuỗi các lớp loại thực thể) mô tả tốt nhất cho chuỗi dữ liệu quan sát (chuỗi các từ) bằng cách tính .

$$P(S \mid O) = \frac{P(S,O)}{P(O)} \tag{2.1}$$

Ở đây S là chuỗi trạng thái ẩn, O là chuỗi dữ liệu quan sát đã biết. Vì P(O) có thể tính được một cách hiệu quả nhờ thuật toán forward-backward [19], việc tìm chuỗi S* làm cực đại xác suất P(S|O) tương đương với việc tìm S* làm cực đại P(S,O).

Ta có thể mô hình hóa HMM dưới dạng một đồ thị có hướng như sau:



Hình 2: Đồ thị có hướng mô tả mô hình HMM

Ở đây, S_i là trạng thái tại thời điểm t=i trong chuỗi trạng thái S, O_i là dữ liệu quan sát được tại thời điểm t=i trong chuỗi O. Sử dụng tính chất Markov thứ nhất (trạng thái hiện tại chỉ phụ thuộc vào trạng thái ngay trước đó) và giả thiết dữ liệu quan sát được tại thời điểm t chỉ phụ thuộc trạng thái tại t, ta có thể tính xác suất P(S,O) như sau:

$$P(S,O) = P(S_1) * P(O_1 \mid S_1) \prod_{t=2}^{n} P(S_t \mid S_{t-1}) * P(O_t \mid S_t) \qquad (2.2)$$

Quá trình tìm ra chuỗi trạng thái tối ưu mô tả tốt nhất chuỗi dữ liệu quan sát cho trước có thể được thực hiện bởi một kĩ thuật lập trình quy hoạch động sử dụng thuật toán Viterbi [19].

2.2.2. Giới hạn của các mô hình Markov ẩn

Trong bài báo "Maximum Entropy Markov Model for Information Extraction and Segmentation"[5], Adrew McCallum đã đưa ra hai vấn đề mà các mô hình HMM truyền thống nói riêng và các mô hình sinh (generative models) nói chung gặp phải khi gán nhãn cho dữ liệu dạng chuỗi.

Thứ nhất, để có thể tính được xác suất P(S, O) (2.1), thông thường ta phải liệt kê hết các trường hợp có thể của chuỗi S và chuỗi O. Nếu như các chuỗi S có thể liệt kê được vì số lượng các trạng thái là có hạn thì trong một số ứng dụng ta không thể nào liệt kê hết được các chuỗi O vì dữ liệu quan sát là hết sức phong phú và đa dạng. Để giải quyết vấn đề này, HMM phải đưa ra giả thiết về sự độc lập giữa các dữ liệu quan sát, đó là dữ liệu quan sát được tại thời điểm t chỉ phụ thuộc trạng thái tại thời điểm đó. Tuy vậy, với các bài toán gán nhãn cho dữ liệu dạng chuỗi, ta nên đưa ra các phương thức biểu diễn các dữ liệu quan sát mềm dẻo hơn như là biểu diễn dữ liệu quan

sát dưới dạng các thuộc tính (features) không phụ thuộc lẫn nhau. Ví dụ với bài toán phân loại các câu hỏi và câu trả lời trong một danh sách FAQ, các thuộc tính có thể là bản thân các từ hay độ dài của dòng, số lượng các kí tự trắng, dòng hiện tại có viết lùi đầu dòng hay không, số các kí tự không nằm trong bảng chữ cái, các thuộc tính về các chức năng ngữ pháp của chúng… Rõ ràng những thuộc tính này không nhất thiết phải độc lập với nhau.

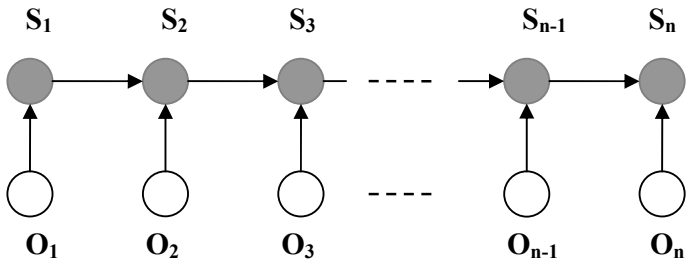
Vấn đề thứ hai mà các mô hình sinh gặp phải khi áp dụng vào các bài toán phân lớp dữ liệu dạng chuỗi đó là chúng sử dụng xác suất đồng thời để mô hình hóa các bài toán có tính điều kiện.Với các bài toán này sẽ thích hợp hơn nếu ta dùng một mô hình điều kiện có thể tính toán P (S|O) trực tiếp thay vì P (S, O) như trong công thức (2.1).

2.3. Mô hình Markov cực đại hóa Entropy (MEMM)

McCallum đã đưa ra một mô hình Markov mới - mô hình MEMM [5] (Maximum Entropy Markov Model) như đáp án cho những vấn đề của mô hình Markov truyền thống.

2.3.1. Tổng quan về mô hình Markov cực đại hóa Entropy (MEMM)

Mô hình MEMM thay thế các xác suất chuyển trạng thái và xác suất sinh quan sát trong HMM bởi một hàm xác suất duy nhất P ($S_i|S_{i-1}$, O_i) - xác suất để trạng thái hiện tại là S_i với điều kiện trạng thái trước đó là S_{i-1} và dữ liệu quan sát hiện tại là O_i. Mô hình MEMM quan niệm rằng các quan sát đã được cho trước và chúng ta không cần quan tâm đến xác suất sinh ra chúng, điều duy nhất cần quan tâm là các xác suất chuyển trạng thái. So sánh với HMM, ở đây quan sát hiện tại không chỉ phụ thuộc vào trạng thái hiện tại mà còn có thể phụ thuộc vào trạng thái trước đó, điều đó có nghĩa là quan sát hiện tại được gắn liền với quá trình chuyển trạng thái thay vì gắn liền với các trạng thái riêng lẻ như trong mô hình HMM truyền thống.



Hình 3: Đồ thị có hướng mô tả một mô hình MEMM

Áp dụng tính chất Markov thứ nhất, xác suất P(S|O) có thể tính theo công thức :

$$P(S \mid O) = P(S_1 \mid O_1) * \prod_{t=1}^{n} P(S_t \mid S_{t-1}, O_1) \quad (2.3)$$

MEMM coi các dữ liệu quan sát là các điều kiện cho trước thay vì coi chúng như các thành phần được sinh ra bởi mô hình như trong HMM vì thế xác suất chuyển trạng thái có thể phụ thuộc vào các thuộc tính đa dạng của chuỗi dữ liệu quan sát. Các thuộc tính này không bị giới hạn bởi giả thiết về tính độc lập như trong HMM và giữ vai trò quan trọng trong việc xác định trạng thái kế tiếp.

Kí hiệu $P_{Si-1}(S_i|O_i)=P(S_i|S_{i-1},O_i)$. Áp dụng phương pháp cực đại hóa Entropy (sẽ được đề cập trong chương 3), McCallum xác định phân phối cho xác suất chuyển trạng thái có dạng hàm mũ như sau:

$$P_{S_{i-1}}(S_i \mid O_i) = \frac{1}{Z(O_i, S_{i-1})} \exp\left(\sum_a \lambda_a f_a(O_i, S_i) \right) \quad (2.4)$$

Ở đây, λ_a là các tham số cần được huấn luyện (ước lượng); Z (O_i, S_i) là thừa số chẩn hóa để tổng xác suất chuyển từ trạng thái S_{i-1} sang tất cả các trạng thái S_i kề đều bằng 1; f_a (O_i, S_i) là hàm thuộc tính tại vị trí thứ i trong chuỗi dữ liệu quan sát và trong chuỗi trạng thái. Mỗi hàm thuộc tính f_a (O_i,S_i) nhận hai tham số, một là dữ liệu quan sát hiện tại O_i và một là trạng thái hiện tại S_i. McCallum định nghĩa a=<b, S_i>, ở đây b là thuộc tính nhị phân chỉ phụ thuộc vào dữ liệu quan sát hiện tại và S_i là trạng thái hiện tại. Sau đây là một ví dụ về một thuộc tính b:

$$b(O_i) = \begin{cases} 1 \text{ nếu dữ liệu quan sát hiện tại là “the”} \\ 0 \text{ nếu ngược lại} \end{cases}$$

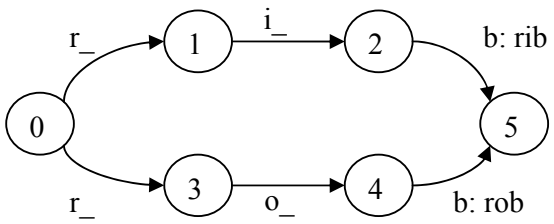
Hàm thuộc tính f_a (O_i, S_i) xác định nếu b (O_i) xác định và trạng thái hiện tại nhận một giá trị cụ thể nào đó:

$$f_a(O_i,S_i) = \begin{cases} 1 \text{ nếu } b(O_i) = 1 \text{ và } S_i = S_{i-1} \\ 0 \text{ nếu ngược lại} \end{cases}$$

Để gán nhãn cho dữ liệu, MEMM xác định chuỗi trạng thái S làm cực đại P(S|O) trong công thức (2.3).Việc xác định chuỗi S cũng được thực hiện bằng cách áp dụng thuật toán Viterbi như trong HMM.

2.3.2. Vấn đề "label bias"

Trong một số trường hợp đặc biệt, các mô hình MEMM và các mô hình định nghĩa một phân phối xác suất cho mỗi trạng thái có thể gặp phải vấn đề "label bias" [15][17]. Ta hãy xem xét một kịch bản chuyển trạng thái đơn giản sau:



Hình 4: Vấn đề "label bias"

Giả sử ta cần xác định chuỗi trạng thái khi xuất hiện chuỗi quan sát là "rob". Ở đây, chuỗi trạng thái đúng S là '0345' và ta mong đợi xác suất P (0345|rob) sẽ lớn hơn xác suất P(0125|rob).

Áp dụng công thức (2.3), ta có:

$$P(0125|rob) = P(0)*P(1|0, r)*P(2|1, o)*P(5|2, b)$$

Vì tổng các xác suất chuyển từ một trạng thái sang các trạng thái kề với nó bằng 1 nên mặc dù trạng thái 1 chưa bao giờ thấy quan sát 'o' nhưng nó không có cách nào khác là chuyển sang trang thái 2, điều đó có nghĩa là P (2|1, x) =1 với x có thể là một quan sát bất kì. Một cách tổng quát, các trạng thái có phân phối chuyển với entropy thấp (ít đường đi ra) có xu hướng ít chú ý hơn đến quan sát hiện tại.

Lại có P (5|2, b) =1, từ đó suy ra: P (0125|rob) = P(0)*P(1|0,r). Tương tự ta cũng có P (0345|rob)=P (0)*P (3|0,r). Nếu trong tập huấn luyện, từ 'rib' xuất hiện thường xuyên hơn từ 'rob' thì xác suất P(3|0,r) sẽ nhỏ hơn xác suất P(1|0,r), điều đó dẫn đến xác suất P(0345|rob) nhỏ hơn xác suất P(0125|rob), tức là chuỗi trạng thái S=0125 sẽ luôn được chọn dù chuỗi quan sát là 'rib' hay 'rob'.

Năm 1991, Léon Bottou đưa ra hai giải pháp cho vấn đề này.Giải pháp thứ nhất là gộp hai trạng thái 1, 3 và trì hoãn việc rẽ nhánh cho đến khi gặp một quan sát

xác định (cụ thể ở đây là ‘i’ và ‘o’). Đây chính là trường hợp đặc biệt của việc chuyển một automata đa định sang một automata đơn định. Nhưng vấn đề ở chỗ ngay cả khi có thể thực hiện việc chuyển đổi này thì cũng gặp phải sự bùng nổ tổ hợp các trạng thái của automata. Giải pháp thứ hai mà Bottou đưa ra là chúng ta sẽ bắt đầu mô hình với một đồ thị đầy đủ của các trạng thái và để cho thủ tục huấn luyện tự quyết định một cấu trúc thích hợp cho mô hình.Tiếc rằng giải pháp này sẽ làm mất tính đi tính có thứ tự của mô hình, một tính chất rất có ích cho các bài tóan trích chọn thông tin [5].

Một giải pháp đúng đắn hơn cho vấn đề này là xem xét toàn bộ chuỗi trạng thái như một tổng thể và cho phép một số các bước chuyển trong chuỗi trạng thái này đóng vai trò quyết định với việc chọn chuỗi trạng thái. Điều này có nghĩa là xác suất của toàn bộ chuỗi trạng thái sẽ không phải được bảo tồn trong quá trình chuyển trạng thái mà có thể bị thay đổi tại một bước chuyển tùy thuộc vào quan sát tại đó.Trong ví dụ trên, xác suất chuyển tại 1 và 3 có thể có nhiều ảnh hưởng đối với việc ta sẽ chọn chuỗi trạng thái nào hơn xác suất chuyển trạng thái tại 0.

2.4. Tổng kết chương

Chương này giới thiêu các hướng tiếp cận nhằm giải quyết bài toán nhận diện loại thực thể: hướng tiếp cận thủ công, các hướng tiếp cận học máy (HMM và MEMM). Trong khi hướng tiếp cận thủ công có giới hạn là tốn kém về công sức, thời gian và không khả chuyển thì HMM không thể tích hợp các thuộc tính phong phú của chuỗi dữ liệu quan sát vào quá trình phân lớp, và MEMM gặp phải vấn đề “label bias”. Những phân tích, đánh giá với từng phương pháp cho thấy nhu cầu về một mô hình thật sự thích hợp cho việc gán nhãn dữ liệu dạng chuỗi nói chung và bài toán nhận diện các loại thực thể nói riêng.

Chương 3. Conditional Random Field (CRF)

CRF [6][11][12][15][16][17] được giới thiệu lần đầu vào năm 2001 bởi Lafferty và các đồng nghiệp. Giống như MEMM, CRF là mô hình dựa trên xác suất điều kiện, nó có thể tích hợp được các thuộc tính đa dạng của chuỗi dữ liệu quan sát nhằm hỗ trợ cho quá trình phân lớp. Tuy vậy, khác với MEMM, CRF là mô hình đồ thị vô hướng. Điều này cho phép CRF có thể định nghĩa phân phối xác suất của toàn bộ chuỗi trạng thái với điều kiện biết chuỗi quan sát cho trước thay vì phân phối trên mỗi trạng thái với điều kiện biết trạng thái trước đó và quan sát hiện tại như trong các mô hình MEMM. Chính vì cách mô hình hóa như vậy, CRF có thể giải quyết được vấn đề 'label bias'. Chương này sẽ đưa ra định nghĩa CRF, một số phương pháp ước lượng tham số cho các mô hình CRF và thuật tóan Viterbi cải tiến để tìm chuỗi trạng thái tốt nhất mô tả một chuỗi dữ liệu quan sát cho trước.

Một số qui ước kí hiệu:

- Chữ viết hoa X, Y, Z…kí hiệu các biến ngẫu nhiên.
- Chữ thường đậm **x**, **y**, **t**, **s**,…kí hiệu các vector như vector biểu diễn chuỗi các dữ liệu quan sát, vector biểu diễn chuỗi các nhãn …
- Chữ viết thường in đậm và có chỉ số là kí hiệu của một thành phần trong một vector, ví dụ $\mathbf{x}_i$ chỉ một thành phần tại vị trí i trong vector x.
- Chữ viết thường không đậm như x, y,… là kí hiệu các giá trị đơn như một dữ liệu quan sát hay một trạng thái.
- *S*: Tập hữu hạn các trạng thái của một mô hình CRF.

3.1. Định nghĩa CRF

Kí hiệu X là biến ngẫu nhiên nhận giá trị là chuỗi dữ liệu cần phải gán nhãn và Y là biến ngẫu nhiên nhận giá trị là chuỗi nhãn tương ứng. Mỗi thành phần Y_i của Y là một biến ngẫu nhiên nhận gía trị trong tập hữu hạn các trạng thái S. Trong bài toán nhận biết các loại thực thể, X có thể nhận giá trị là các câu trong ngôn ngữ tự nhiên, Y là một chuỗi ngẫu nhiên các tên thực thể tương ứng với các câu này và mỗi một thành phần Y_i của Y có miền giá trị là tập tất cả các nhãn tên thực thể (tên người, tên địa danh,...).

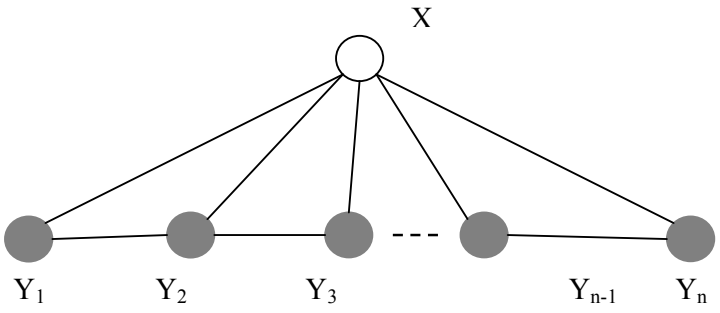
Cho một đồ thị vô hướng không có chu trình G=(V,E), ở đây V là tập các đỉnh của đồ thị và E là tập các cạnh vô hướng nối các đỉnh đồ thị. Các đỉnh V biểu diễn các thành phần của biến ngẫu nhiên Y sao cho tồn tại ánh xạ một-một giữa một đỉnh và

một thành phần của Y_v của Y. Ta nói (Y|X) là một trường ngẫu nhiên điều kiện (Conditional Random Field - CRF) khi với điều kiện X, các biến ngẫu nhiên Y_v tuân theo tính chất Markov đối với đồ thị G:

$$P(Y_v \mid X, Y_\omega, \omega \neq v) = P(Y_v \mid X, Y_\omega, \omega \in N(v)) \quad (3.1)$$

Ở đây, N(v) là tập tất cả các đỉnh kề với v. Như vậy, một CRF là một trường ngẫu nhiên phụ thuộc tòan cục vào X. Trong các bài toán xử lý dữ liệu dạng chuỗi, G đơn giản chỉ là dạng chuỗi G=(V={1,2,…m},E={(i,i+1)}).

Kí hiệu X=(X_1, X_2,…, X_n), Y=(Y_1,Y_2, ...,Y_n). Mô hình đồ thị cho CRF có dạng:



Hình 5: Đồ thị vô hướng mô tả CRF

Gọi C là tập hợp tất cả các đồ thị con đầy đủ của đồ thị G - đồ thị biểu diễn cấu trúc của một CRF. Áp dụng kết quả của Hammerley-Clifford [14] cho các trường ngẫu nhiên Markov, ta thừa số hóa được p(**y**|**x**) - xác suất của chuỗi nhãn với điều kiện biết chuỗi dữ liệu quan sát- thành tích của các hàm tiềm năng như sau:

$$P(\mathbf{y} \mid \mathbf{x}) = \prod_{A \in C} \psi_A(A \mid \mathbf{x}) \quad (3.2)$$

Vì trong các bài toán xử lý dữ liệu dạng chuỗi đồ thị biểu diễn cấu trúc của một CRF có dạng đường thẳng như trong hình 5 nên tập C phải là hợp của E và V, trong đó E là tập các cạnh của đồ thị G và V là tập các đỉnh của G, hay nói cách khác đồ thị con A hoặc chỉ gồm một đỉnh hoặc chỉ gồm một cạnh của G.

3.2. Nguyên lý cực đại hóa Entropy

Lafferty et. al.[17] xác định các hàm tiềm năng cho các mô hình CRF dựa trên nguyên lý cực đại hóa Entropy [1][3][8][29]. Cực đại hóa Entropy là một nguyên lý cho phép đánh giá các phân phối xác suất từ một tập các dữ liệu huấn luyện.

3.2.1. Độ đo Entropy điều kiện

Entropy là độ đo về tính đồng đều hay tính không chắc chắn của một phân phối xác suất. Độ đo Entropy điều kiện của một phân phối mô hình trên "một chuỗi trạng thái với điều kiện biết một chuỗi dữ liệu quan sát" p(**y**|**x**) có dạng sau:

$$H(p) = -\sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x}) * p(\mathbf{y} \mid \mathbf{x}) * \log p(\mathbf{y} \mid \mathbf{x}) \tag{3.3}$$

3.2.2. Các ràng buộc đối với phân phối mô hình

Các ràng buộc đối với phân phối mô hình được thiết lập bằng cách thống kê các thuộc tính được rút ra từ tập dữ liệu huấn luyện. Dưới đây là ví dụ về một thuộc tính như vậy:

$$\text{f} = \begin{cases} 1 \text{ nếu từ liền trước là từ "ông" và nhãn hiện tại là B\_PER} \\ 0 \text{ nếu ngược lại} \end{cases}$$

Tập các thuộc tính là tập hợp các thông tin quan trọng trong dữ liệu huấn luyện. Kí hiệu kì vọng của thuộc tính f theo phân phối xác suất thực nghiệm như sau:

$$E_{\tilde{p}(\mathbf{x},\mathbf{y})}[f] \equiv \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x},\mathbf{y}) f(\mathbf{x},\mathbf{y}) \tag{3.4}$$

Ở đây $\tilde{p}(\mathbf{x},\mathbf{y})$ là phân phối thực nghiệm trong dữ liệu huấn luyện. Giả sử dữ liệu huấn luyện gồm N cặp, mỗi cặp gồm một chuỗi dữ liệu quan sát và một chuỗi nhãn D={($\mathbf{x}^i$,$\mathbf{y}^i$)}, khi đó phân phối thực nghiệm trong dữ liệu huấn luyện được tính như sau:

$\tilde{p}(\mathbf{x},\mathbf{y})$=1/N * số lần xuất hiện đồng thời của **x,y** trong tập huấn luyện

Kì vọng của thuộc tính f theo phân phối xác suất trong mô hình

$$E_p[f] \equiv \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x}) * p(\mathbf{y} \mid \mathbf{x}) * f(\mathbf{x},\mathbf{y}) \tag{3.5}$$

Phân phối mô hình thống nhất với phân phối thực nghiệm chỉ khi kì vọng của mọi thuộc tính theo phân phối xác suất phải bằng kì vọng của thuộc tính đó theo phân phối mô hình :

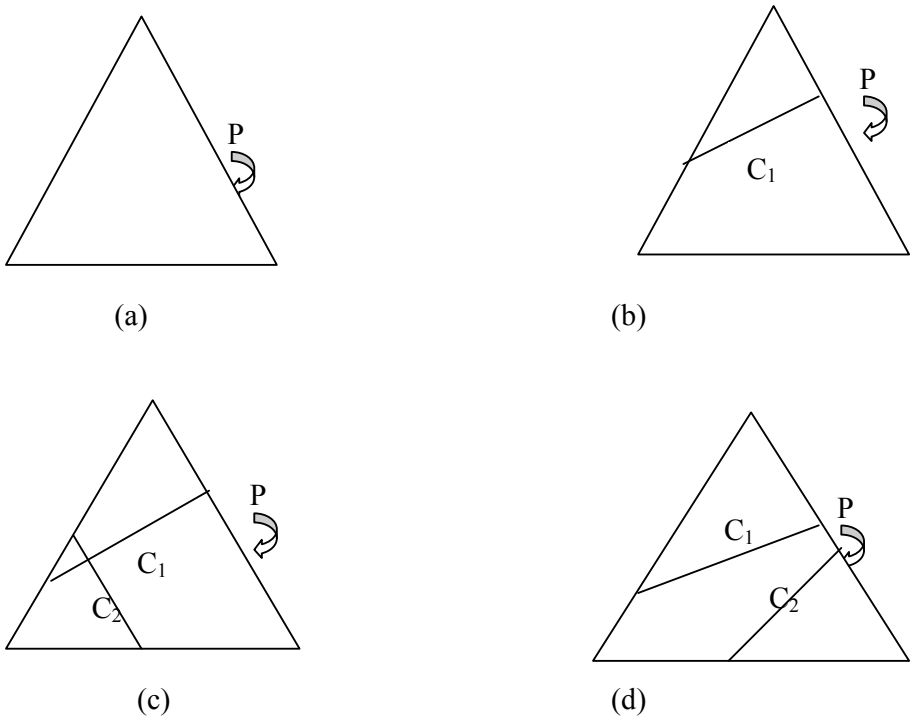
$$E_{\tilde{p}(\mathbf{x},\mathbf{y})}[f] = E_p[f] \tag{3.6}$$

Phương trình (3.6) thể hiện một ràng buộc đối với phân phối mô hình. Nếu ta chọn n thuộc tính từ tập dữ liệu huấn luyện, ta sẽ có tương đương n ràng buộc đối với phân phối mô hình.

3.2.3. Nguyên lý cực đại hóa Entropy

Gọi P là không gian của tất cả các phân phối xác suất điều kiện, và n là số các thuộc tính rút ra từ dữ liệu huấn luyện. P' là tập con của P, P' được xác định như sau:

$$P' = \left\{ p \in P \mid E_p(f_i) = E_{\tilde{p}}(f_i) \forall i \in \{1,2,3...,n\} \right\} \quad (3.7)$$



(a) (b) (c) (d)

Hình 6: Các ràng buộc mô hình

P là không gian của toàn bộ phân phối xác suất. Trường hợp a: không có ràng buộc; trường hợp b: có một ràng buộc C_1, các mô hình p thỏa mãn ràng buộc nằm trên đường C_1; trường hợp c: 2 ràng buộc C_1 và C_2 giao nhau, mô hình p thỏa mãn cả hai ràng buộc là giao của hai đường C_1 và C_2; trường hợp d: 2 ràng buộc C_1 và C_2 không giao nhau, không tồn tại mô hình p thỏa mãn cả 2 ràng buộc.

Tư tưởng chủ đạo của nguyên lý cực đại hóa Entropy là ta phải xác định một phân phối mô hình sao cho "phân phối đó tuân theo mọi giả thiết đã biết từ thực

nghiệm và ngoài ra không đưa thêm bất kì một giả thiết nào khác". Điều này có nghĩa là phân phối mô hình phải thỏa mãn mọi ràng buộc được rút ra từ thực nghiệm, và phải gần nhất với phân phối đều. Nói theo ngôn ngữ toán học, ta phải tìm phân phối mô hình p(**y**|**x**) thỏa mãn hai điều kiện, một là nó phải thuộc tập P' (3.7) và hai là nó phải làm cực đại Entropy điều kiện (3.3).

Với mỗi thuộc tính f_i ta đưa vào một thừa số langrange λ_i, ta định nghĩa hàm Lagrange $L(p,\lambda)$ như sau:

$$L(p,\lambda) = H(p) + \sum_i \lambda_i * (E_{\tilde{p}}[f_i] - E_p[f_i]) \tag{3.8}$$

Phân phối p(**y**|**x**) làm cực đại độ đo Entropy $H(p)$ và thỏa mãn n ràng buộc dạng $E_{\tilde{p}(\mathbf{x},\mathbf{y})}[f] = E_p[f]$ cũng sẽ làm cực đại hàm $L(p,\lambda)$ (theo lý thuyết thừa số Langrange). Từ (3.8) ta suy ra:

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z_\lambda(\mathbf{x})} \exp\left(\sum_i \lambda_i f_i\right) \tag{3.9}$$

Ở đây $Z_\lambda(\mathbf{x})$ là thừa số chuẩn hóa để đảm bảo $\sum_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}) = 1$ với mọi **x:**

$$Z_\lambda(\mathbf{x}) = \sum_{\mathbf{y}} \exp\left(\sum_i \lambda_i f_i\right) \tag{3.10}$$

3.3. Hàm tiềm năng của các mô hình CRF

Bằng cách áp dụng nguyên lý cực đại hóa Entropy, Lafferty xác định hàm tiềm năng của một CRF có dạng một hàm mũ.

$$\psi_A(A \mid \mathbf{x}) = \exp \sum_k \gamma_k f_k(A \mid \mathbf{x}) \tag{3.11}$$

Ở đây f_k là một thuộc tính của chuỗi dữ liệu quan sát và γ_k là trọng số chỉ mức độ biểu đạt thông tin của thuộc tính f_k .

Có hai loại thuộc tính là thuộc tính chuyển (kí hiệu là t) và thuộc tính trạng thái(kí hiệu là s) tùy thuộc vào A là đồ thị con gồm một đỉnh hay một cạnh của G. Thay các hàm tiềm năng vào công thức (3.2) và thêm vào đó một thừa số chuẩn hóa Z(x) để đảm bảo tổng xác suất của tất cả các chuỗi nhãn tương ứng với một chuỗi dữ liệu quan sát bằng 1, ta được:

$$P(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_i \sum_k \lambda_k t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x}) + \sum_i \sum_k \mu_k s_k(\mathbf{y}_i, \mathbf{x}) \right) \quad (3.12)$$

Ở đây, **x**,**y** là chuỗi dữ liệu quan sát và chuỗi trạng thái tương ứng; t_k là thuộc tính của tòan bộ chuỗi quan sát và các trạng thái tại ví trí i-1, i trong chuỗi trạng thái; s_k là thuộc tính của toàn bộ chuỗi quan sát và trạng thái tại ví trí i trong chuỗi trạng thái.

$$s_i = \begin{cases} 1 \text{ nếu } \mathbf{x_i}\text{=Bill và } \mathbf{y_i}\text{= B\_PER} \\ 0 \text{ nếu ngược lại} \end{cases}$$

$$t_i = \begin{cases} 1 \text{ nếu } \mathbf{x_{i-1}}\text{= “Bill”, } \mathbf{x_i}\text{=”Clinton” và } \mathbf{y_{i-1}}\text{=B\_PER,}\mathbf{y_i}\text{=I\_PER} \\ 0 \text{ nếu ngược lại} \end{cases}$$

Thừa số chuẩn hóa Z(**x**) được tính như sau:

$$Z(\mathbf{x}) = \sum_y \exp\left(\sum_i \sum_k \lambda_k t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x}) + \sum_i \sum_k \mu_k s_k(\mathbf{y}_i, \mathbf{x}) \right) \quad (3.13)$$

$\theta(\lambda_1, \lambda_2, ..., \mu_1, \mu_2 ..)$ là các vector các tham số của mô hình, teta sẽ được ước lượng giá trị nhờ các phương pháp ước lượng tham số cho mô hình sẽ được đề cập trong phần sau.

3.4. Thuật toán gán nhãn cho dữ liệu dạng chuỗi

Tại mỗi vị trí i trong chuỗi dữ liệu quan sát, ta định nghĩa một ma trận chuyển |*S*|*|*S*| như sau:

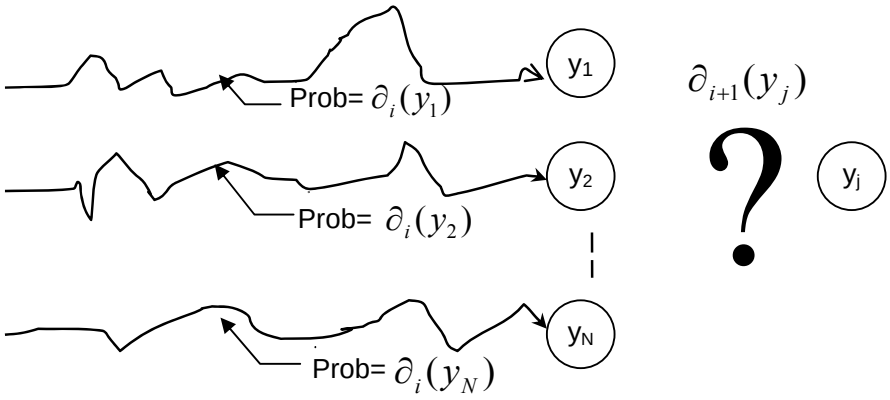
$$M_i(\mathbf{x}) = \left[M_i(y', y, \mathbf{x}) \right] \quad (3.14)$$

$$M_i(y', y, \mathbf{x}) = \exp\left(\sum_k \lambda_k t_k(y', y, \mathbf{x}) + \sum_k \mu_k s_k(y, \mathbf{x}) \right) \quad (3.15)$$

Ở đây M_i(y’,y,**x**) là xác suất chuyển từ trạng thái y’ sang trạng thái y với chuỗi dữ liệu quan sát là **x**. Chuỗi trạng thái **y*** mô tả tốt nhất cho chuỗi dữ liệu quan sát **x** là nghiệm của phương trình:

$$\mathbf{y}^* = \text{argmax}\{p(\mathbf{y}|\mathbf{x})\} \quad (3.16)$$

Chuỗi **y*** được xác định bằng thuật toán Viterbi cải tiến. Định nghĩa $\partial_i(y)$ là xác suất của "chuỗi trạng thái độ dài i kết thúc bởi trạng thái y và có xác suất lớn nhất" biết chuỗi quan sát là **x**.



Hình 7: Một bước trong thuật toán Viterbi cải tiến

Giả sử biết tất cả $\partial_i(y_k)$ với mọi y_k thuộc tập trạng thái *S* của mô hình, cần xác định $\partial_{i+1}(y_j)$. Từ hình 7, ta suy ra công thức đệ quy

$$\partial_{i+1}(y_j) = \max\left(\partial_{i-1}(y_k) * M_i(y_k, y_j, \mathbf{x})\right) \forall y_k \in S \qquad (3.17)$$

Đặt $\Pr e_i(y) = \arg\max\left(\partial_{i-1}(y') * M_i(y', y, x)\right)$. Giả sử chuỗi dữ liệu quan sát **x** có độ dài n, sử dụng kĩ thuật backtracking để tìm chuỗi trạng thái **y*** tương ứng như sau:

- Bước 1: Với mọi y thuộc tập trạng thái tìm
  - $\mathbf{y}^*(n) = \arg\max\left(\partial_n(y)\right)$
  - i ← n
- Bước lặp: chừng nào i>0
  - i ← i-1
  - y ← Pre_i(y)
  - **y***(i) = y

Chuỗi **y*** tìm được 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 cho trước.

3.5. CRF có thể giải quyết được vấn đề 'label bias'

Bản chất phân phối toàn cục của CRF giúp cho các mô hình này tránh được vấn đề 'label bias' được miêu tả trong phần 2.3.2 trên đây. Ở phương diện lý thuyết mô hình, ta có thể coi mô hình CRF như là một máy trạng thái xác suất với các trọng số không chuẩn hóa, mỗi trọng số gắn liền với một bước chuyển trạng thái. Bản chất không chuẩn hóa của các trọng số cho phép các bước chuyển trạng thái có thể nhận các giá trị quan trọng khác nhau. Vì thế bất cứ một trạng thái nào cũng có thể làm tăng hoặc giảm xác suất được truyền cho các trạng thái sau nó 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.

Trong [17], Lafferty và các đồng nghiệp của ông đã tiến hành thử nghiệm với 2000 mẫu dữ liệu huấn luyện và 500 mẫu kiểm tra, các mẫu này đều chứa các trường hợp nhập nhằng như trong ví dụ miêu tả ở phần 2.3.2. Thực nghiệm cho thấy tỉ lệ lỗi của CRF là 4.6% trong khi tỉ lệ lỗi của MEMM là 42%, điều này chứng tỏ rằng các mô hình MEMM không xác định được nhánh rẽ đúng trong trường hợp 'label bias'

3.6. Tổng kết chương

Chương này giới thiệu những vấn đề cơ bản về CRF: định nghĩa CRF, thuật toán gán nhãn cho dữ liệu dạng chuỗi trong CRF, nguyên lý cực đại hóa Entropy để xác định các hàm tiềm năng cho các mô hình CRF, chứng minh CRF có thể giải quyết được vấn đề 'label bias'. Áp dụng các mô hình CRF trong các bài toán xử lý dữ liệu chuỗi [5] [9] cho thấy CRF có khả năng xử lý dữ liệu dạng này mạnh hơn so với các mô hình học máy khác như HMM hay MEMM.

Chương 4. Ước lượng tham số cho các mô hình CRF

Kĩ thuật được sử dụng để đánh giá tham số cho một mô hình CRF là làm cực đại hóa độ đo likelihood giữa phân phối mô hình và phân phối thực nghiệm.

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=$\{(\mathbf{x}^{(i)},\mathbf{y}^{(i)})\}$ $\forall i = 1 \ldots N$. Độ đo likelihood giữa tập huấn luyện và mô hình điều kiện tương ứng p($\mathbf{y}|\mathbf{x},\theta$) là:

$$L(\theta) = \prod_{\mathbf{x},\mathbf{y}} p(\mathbf{y} \mid \mathbf{x},\theta)^{\tilde{p}(\mathbf{x},\mathbf{y})} \tag{4.1}$$

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.

Nguyên lý cực đại likelihood: các tham số tốt nhất của mô hình là các tham số làm cực đại hàm likelihood.

$$\theta_{ML} = \arg\max_{\theta} L(\theta) \tag{4.2}$$

θ_{ML} đảm bảo những dữ liệu mà chúng ta quan sát được trong tập huấn luyện sẽ nhận được xác suất cao trong mô hình. Nói cách khác, các tham số làm cực đại hàm likelihood sẽ làm phân phối trong mô hình gần nhất với phân phối thực nghiệm trong tập huấn luyện. Vì việc tính teta dựa theo công thức (4.1) rất khó khăn nên thay vì tính toán trực tiếp, ta đi xác định teta làm cực đại logarit của hàm likelihood (thường được gọi tắt là log-likelihood):

$$l(\theta) = \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x},\mathbf{y}) \log\big(p(\mathbf{y} \mid \mathbf{x},\theta)\big) \tag{4.3}$$

Vì hàm logarit là hàm đơn điệu nên việc làm này không làm thay đổi giá trị của θ được chọn.Thay p($\mathbf{y}|\mathbf{x},\theta$) của mô hình CRF vào công thức (4.3), ta có:

$$l(\theta) = \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x},\mathbf{y}) \left[\sum_{i=1}^{n+1} \lambda * \mathbf{t} + \sum_{i=1}^{n} \mu * \mathbf{s} \right] - \sum_{\mathbf{x}} \tilde{p}(\mathbf{x}) * \log Z \tag{4.4}$$

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, **t** là vector các thuộc tính chuyển ($t_1(y_{i-1},y_i,x),t_2(y_{i-1},y_i,x),\ldots$), **s** là vector các thuộc tính trạng thái ($s_1(y_i,x),s_2(y_i,x),\ldots$).

Hàm log-likelihood cho mô hình CRF là một hàm lõm và trơn trong toàn bộ không gian của tham số. Bản chất hàm lõm của log-likelihood cho phép ta có thể tìm được giá trị cực đại toàn cục θ bằng cách thiết lập các thành phần của vector gradient của hàm log-likelihood bằng không. Mỗi thành phần trong vector gradient của hàm log-likelihood là đạo hàm của hàm log-likelihood theo một tham số của mô hình. Đạo hàm hàm log – likelihood theo tham số λ_k ta được:

$$\frac{\partial l(\theta)}{\partial \lambda_k} = \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x},\mathbf{y}) \sum_{i=1}^{n} t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x})$$

$$- \sum_{\mathbf{x}} \tilde{p}(\mathbf{x}) p(\mathbf{y} \mid \mathbf{x}, \theta) \sum_{i=1}^{n} t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x})$$

$$= E_{\tilde{p}(\mathbf{x},\mathbf{y})}[t_k] - E_{p(\mathbf{y}|\mathbf{x},\theta)}[t_k] \qquad (4.5)$$

Việc thiết lập phương trình trên bằng 0 tương đương với việc đưa ra một ràng buộc cho mô hình: giá trị trung bình của t_k theo phân phối $\tilde{p}(\mathbf{x})p(\mathbf{y} \mid \mathbf{x}, \theta)$ bằng giá trị trung bình của t_k theo phân phối thực nghiệm $\tilde{p}(\mathbf{x},\mathbf{y})$.

Về phương diện toán học, bài toán ước lượng tham số cho một mô hình CRF chính là bài toán tìm cực đại của hàm log-likelihood. Chương này giới thiệu một số phương pháp tìm cực đại của log-likelihood: các phương pháp lặp (IIS và GIS), các phương pháp tối ưu số (Conjugate Gradient, các phương pháp Newton...).

4.1. Các phương pháp lặp

Các phương pháp lặp làm mịn dần phân phối mô hình bằng các cập nhật các tham số mô hình theo cách

$$\lambda_k \leftarrow \lambda_k + \delta\lambda_k \qquad (4.6)$$

Ở đây, các giá trị $\partial\lambda_k$ được chọn sao cho giá trị của hàm likelihood gần với cực đại hơn. Lafferty et. al. [17] đưa ra hai thuật toán lặp cho việc ước lượng tham số cho mô hình CRF, một là IIS và một là GIS. Trong phần này, chúng ta sẽ tìm hiểu về phương pháp lặp tổng quát sau đó đi sâu tìm hiểu hai thuật toán IIS và GIS.

Giả sử chúng ta có một mô hình $p(\mathbf{y} \mid \mathbf{x}, \theta)$ ở đây $\theta(\lambda_1, \lambda_2, ..., \mu_1, \mu_2, ...)$, mục đích của các phương pháp lặp là tìm một tập các tham số mới $\theta + \Delta$ sao cho hàm log-likelihood nhận giá trị lớn hơn với tập tham số cũ, ở đây $\Delta = (\delta\lambda_1, \delta\lambda_2, ..., \delta\mu_1, \delta\mu_2, ...)$. Nói cách khác, trong các phương pháp lặp ta phải tìm một cách thức cập nhật tham số

mô hình sao cho hàm log-likelihood nhận giá trị càng gần với giá trị cực đại càng tốt. Việc cập nhật tham số sẽ được lặp lại cho đến khi hàm log-likelihood hội tụ (gia số của hàm log-likelhood có trị tuyệt đối nhỏ hơn một giá trị ε nào đó). Với mô hình CRF, gia số của hàm log-likelihood bị chặn dưới bởi một hàm phụ $A(\theta,\Delta)$ được định nghĩa như sau

$$A(\theta,\Delta) \equiv \sum_{\mathbf{x},\mathbf{y}}\left[\sum_{i=1}^{n+1}\sum_{k}\lambda_k t_k(\mathbf{y}_{i-1},\mathbf{y}_i,\mathbf{x}) + \sum_{i=1}^{n}\sum_{k}\mu_k s_k(\mathbf{y}_i,\mathbf{x})\right]$$
$$+ 1 - \sum \tilde{p}(\mathbf{x})p(\mathbf{y}\mid\mathbf{x},\theta)\left[\sum_{t=1}^{n+1}\sum_{k}\left(\frac{t_k(\mathbf{y}_{i-1},\mathbf{y}_i,\mathbf{x})}{T(\mathbf{x},\mathbf{y})}\right)\exp\left(\delta\lambda_k T(\mathbf{x},\mathbf{y})\right)\right.$$
$$\left. + \sum_{i=1}^{n}\sum_{k}\frac{s_k(\mathbf{y}_i,\mathbf{x})}{T(\mathbf{x},\mathbf{y})}\exp\left(\delta\mu_k T(\mathbf{x},\mathbf{y})\right)\right] \qquad (4.7)$$

Ở đây $T(\mathbf{x},\mathbf{y})$ là tổng các thuộc tính của chuỗi dữ liệu quan sát và chuỗi các nhãn tương ứng (**x,y**)

$$T(\mathbf{x},\mathbf{y}) \equiv \sum_{i=1}^{n+1}\sum_{k}t_k(\mathbf{y}_{i-1},\mathbf{y}_i,\mathbf{x}) + \sum_{i=1}^{n}\sum_{k}s_k(\mathbf{y}_i,\mathbf{x}) \qquad (4.8)$$

Vì $l(\theta+\Delta) - l(\theta) \geq A(\theta,\Delta)$ nên Δ làm cực đại $A(\theta,\Delta)$ cũng sẽ làm cực đại gia số của hàm log-likelihood. Dưới đây là thủ tục lặp để tìm tập tham số làm cực đại hàm likelihood.

- Khởi tạo các λ_k
- Lặp cho đến khi nào hội tụ
  - Giải phương trình $\frac{\partial A(\theta,\Delta)}{\partial\delta\lambda_k} = 0$ với mỗi tham số λ_k
  - Cập nhật các tham số $\lambda_k \leftarrow \lambda_k + \delta\lambda_k$

Thiết lập đạo hàm từng phần của $A(\theta,\Delta)$ theo tham số λ_k bằng không ta thu được phương trình sau:

$$E_{\tilde{p}(\mathbf{x},y)}[t_k] \equiv \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x},\mathbf{y}) \sum_{i=1}^{n+1} t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x}) \tag{4.9}$$

$$= \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x}) p(\mathbf{y} \mid \mathbf{x}, \theta) \sum_{i=1}^{n+q} t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x}) \exp(\delta\lambda_k T(\mathbf{x},\mathbf{y})) \tag{4.10}$$

Từ đây, ta có thể tính được các gia số $\delta\lambda_k$ và $\delta\mu_k$. IIS [2][15] và GIS [15] là hai trường hợp đặc biệt của phương pháp lặp, mỗi thuật toán có một cách chọn vector gia số để cập nhật tham số khác nhau.

4.1.1. Thuật toán GIS

Đặt C là giá trị lớn nhất của $T(\mathbf{x},\mathbf{y})$ với tất cả **x,y** trong tập dữ liệu huấn luyện. Định nghĩa một vector thuộc tính toàn cục (thuộc tính không gắn liền với một cạnh hay một đỉnh nào trong đồ thị mô tả một CRF) .

$$g(\mathbf{x},\mathbf{y}) \equiv C - \sum_{i=1}^{n+1} \sum_{k} t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x}) + \sum_{i=1}^{n} \sum_{k} s_k(\mathbf{y}_i, \mathbf{x}) \tag{4.11}$$

Thông thường việc thêm vào một thuộc tính sẽ làm thay đổi phân phối xác suất của mô hình, tuy nhiên các thuộc tính toàn cục g(**x,y)** hòan toàn phụ thuộc vào các thuộc tính đã có trong mô hình, điều này có nghĩa là ta không đưa thêm một ràng buộc nào đối với phân phối mô hình hay nói cách khác phân phối mô hình sẽ không đổi khi thêm vào thuộc tính toàn cục. Mặc dù không làm thay đổi phân phối mô hình, việc thêm các thuộc tính g(**x,y**) lại làm thay đổi giá trị của $T(\mathbf{x},\mathbf{y})$, tính cả các thuộc tính toàn cục $T(\mathbf{x},\mathbf{y})$ sẽ luôn nhận giá trị hằng số C. Nếu các thuộc tính chỉ nhận gía trị 0,1 thì T(**x,y**) sẽ chính là số các thuộc tính hoạt động trong mô hình.

Với giả thiết $T(\mathbf{x},\mathbf{y})=C$, Lafferty et.al [15][17] chứng minh rằng phương trình (4.10) có thể giải theo phương pháp giải tích thông thường. Logarithm hai vế của phương trình (4.10), ta có:

$$\log E_{\tilde{p}(\mathbf{x},\mathbf{y})}[t_k] = \log\left[\sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x}) p(\mathbf{y} \mid \mathbf{x}, \theta) \sum_{i=1}^{n+1} t_k(\mathbf{y}_{i-1}, \mathbf{y}_i, \mathbf{x}) \exp(\delta\lambda_k * C) \right]$$

$$= \log E_{p(\mathbf{y}|\mathbf{x},\theta)}[t_k] + \delta\lambda_k * C \tag{4.12}$$

Từ đây, suy ra:

$$\delta\lambda_k = \frac{1}{C}\log\left[\frac{E_{\tilde{p}(\mathbf{x},\mathbf{y})}[t_k]}{E_{p(\mathbf{x},\mathbf{y})}[t_k]}\right] \tag{4.13}$$

Tốc độ hội tụ của thuật toán GIS phụ thuộc độ lớn của *C*, *C* càng lớn các bước cập nhật càng nhỏ, tỉ lệ hội tụ càng chậm, ngược lại *C* càng nhỏ, tốc độ hội tụ càng nhanh.

4.1.2. Thuật toán IIS

Tư tưởng của thuật toán IIS: biểu diễn phương trình (4.10) dưới dạng một đa thức của $\exp(\delta\lambda_k)$, áp dụng phương pháp Newton-Raphson giải đa thức nhận được để tìm $\delta\lambda_k$.

Để biểu diễn phương trình (4.10) dưới dạng đa thức của $\exp(\delta\lambda_k)$, Lafferty et.al đưa ra xấp xỉ

$$T(\mathbf{x},\mathbf{y}) \approx T(\mathbf{x}) = \max_{\mathbf{y}} T(\mathbf{x},\mathbf{y}) \tag{4.14}$$

Thay $T(\mathbf{x},\mathbf{y})$ vào phương trình (4.10), ta có:

$$E_{\tilde{p}(\mathbf{x},\mathbf{y})}[t_k] = \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x})\, p(\mathbf{y}\mid\mathbf{x},\theta) \sum_{i=1}^{n+1} t_k(\mathbf{y}_{i-1},\mathbf{y}_i,\mathbf{x}) \exp(\delta\lambda_k T(\mathbf{x})) \tag{4.15}$$

Phân hoạch tập các cặp (**x**,**y**) thành $T_{\max}$ tập con không giao nhau, ở đây $T_{\max} = \max T(\mathbf{x})$. Viết lại (4.15) dưới dạng

$$E_{\tilde{p}(\mathbf{x},\mathbf{y})}[t_k] = \sum_{m=0}^{T_{\max}} \sum_{\{\mathbf{x},\mathbf{y}\mid T(\mathbf{x})=m\}} \tilde{p}(\mathbf{x})\, p(\mathbf{y}\mid\mathbf{x},\theta) \sum_{i=1}^{n+1} t_k(\mathbf{y}_{i-1},\mathbf{y}_i,\mathbf{x}) \left[\exp(\delta\lambda_k)\right]^m \tag{4.16}$$

Định nghĩa $a_{k,m}$ là kì vọng của t_k trong tập các cặp (**x**,**y**) có $T(\mathbf{x}) = m$.

$$a_{k,m} = \sum_{\mathbf{x},\mathbf{y}} \tilde{p}(\mathbf{x})\, p(\mathbf{y}\mid\mathbf{x},\theta) \sum_{i=1}^{n+1} t_k(\mathbf{y}_{i-1},\mathbf{y}_i,\mathbf{x}) \delta(m, T(\mathbf{x})) \tag{4.17}$$

Ở đây, $\delta(m, T(\mathbf{x}))$ được định nghĩa như sau:

$$\delta(m, T(\mathbf{x})) = \begin{cases} 1 \text{ nếu } T(\mathbf{x})=\text{m} \\ 0 \text{ nếu ngược lại} \end{cases}$$

Khi đó, phương trình (4.16) có thể viết lại dưới dạng

$$E_{\tilde{p}(\mathbf{x},\mathbf{y})}[t_k] = \sum_{m=0}^{T_{\max}} a_{k,m} [\exp(\delta\lambda_k)]^m \tag{4.18}$$

Giải phương trình (4.18) theo phương pháp Newton-Raphson ta tìm được $\delta\lambda_k$.

4.2. Các phương pháp tối ưu số

Các kĩ thuật tối ưu số[15][28] sử dụng vector gradient của hàm log-likelihood để tìm cực trị. Hai loại kĩ thuật tối ưu được đề cập trong phần nay là kĩ thuật tối ưu bậc một và kĩ thuật tối ưu bậc hai.

4.2.1. Kĩ thuật tối ưu số bậc một

Kĩ thuật tối ưu số bậc một sử dụng các thông tin chứa trong bản thân vector gradient của hàm cần tối ưu để dần dần tịnh tiến các ước lượng đến điểm mà vector gradient bằng 0 và hàm đạt cực trị. Có hai phương pháp tối ưu bậc một có thể dùng để ước lượng tham số cho một mô hình CRF, cả hai phương pháp này đều là biến thể của thuật toán "gradient liên hợp không tuyến tính" (non-linear conjugate gradient).

Không xem xét một hướng tìm kiếm trong khi làm cực đại hàm số như các phương pháp leo đồi, các phương pháp "hướng liên hợp" sinh ra một tập các vector khác không – tập liên hợp – và lần lượt làm cực đại hàm dọc theo hướng này. Các phương pháp "gradient liên hợp không tuyến tính" là trường hợp đặc biệt của kĩ thuật hướng liên hợp trong đó mỗi "vector liên hợp" hay "hướng tìm kiếm" chỉ được sinh từ hướng tìm kiếm trước đó mà không phải từ tất cả các thành phần của tập liên hợp trước đó. Đặc biệt, mỗi hướng tìm kiếm p_j sau là tổ hợp tuyến tính của "hướng đi lên dốc nhất" hay gradient của hàm cần tìm cực trị và hướng tìm kiếm trước đó p_{j-1}. Mỗi bước lặp của thuật tóan cập nhật gradient liên hợp tịnh tiến các tham số của hàm cần tìm cực đại theo hướng của vector liên hợp hiện thời sử dụng luật cập nhật:

$$\lambda_k^{(j+1)} = \lambda_k^{j} + \alpha^{(j)} p_j \tag{4.19}$$

Ở đây, $\alpha^{(j)}$ là độ lớn của bước nhẩy tối ưu.

Có hai phương pháp tối ưu bậc một rất thích hợp cho việc ước lượng tham số mô hình CRF, đó là các thuật tóan Fletcher-Reeves và Polak-Ribière-Positive. Về bản chất hai thuật toán này là hoàn toàn tương đương, chúng chỉ khác nhau về cách chọn hướng tìm kiếm và độ lớn của bước nhẩy tối ưu.

4.2.2. Kĩ thuật tối ưu số bậc hai

Ngoài giá trị của vector gradient, các kĩ thuật tối ưu số bậc hai cải tiến các kĩ thuật bậc một trong việc tính toán các cập nhật cho tham số bằng cách thêm yếu tố về đường cong hay đạo hàm bậc hai của hàm cần tìm cực trị.

Luật cập nhật bậc hai được tính toán bằng cách khai triển chuỗi Taylor bậc hai của $l(\theta + \Delta)$ như sau:

$$l(\theta + \Delta) \approx l(\theta) + \Delta^T G(\theta) + \frac{1}{2}\Delta^T H(\theta)\Delta \tag{4.20}$$

$G(\theta)$ và $H(\theta)$ lần lượt là vector gradient và ma trận Hessian (ma trận đạo hàm từng phần bậc hai) của hàm log-likelihood $l(\theta)$. Thiết lập đạo hàm của xấp xỉ trong (4.20) bằng 0 ta tìm được gia số để cập nhật tham số mô hình như sau:

$$\Delta^{(k)} = H^{-1}(\theta^{(k)})G(\theta^{(k)}) \tag{4.21}$$

Ở đây, k là chỉ số của lần lặp hiện tại. Mặc dù việc cập nhật các tham số mô hình theo cách thức này cho hội tụ rất nhanh nhưng việc tính nghịch đảo của ma trận Hessian lại đòi hỏi chi phí lớn về thời gian đặc biệt là với các bài toán cỡ lớn như là các bài tóan trong xử lý ngôn ngữ tự nhiên. Vì thế các phương pháp bậc hai mà phải tính tóan trực tiếp nghịch đảo của ma trận Hessian không thích hợp cho việc ước lượng tham số cho các mô hình CRF.

Các phương pháp quasi-Newton là các trường hợp đặc biệt của kĩ thuật tối ưu bậc hai, tương tự như các phương pháp Newton tuy nhiên chúng không tính toán trực tiếp ma trận Hessian mà thay vào đó chúng xây dựng một mô hình của ma trận Hessian tại mỗi bước lặp bằng cách đo độ thay đổi trong vector gradient.

Yếu tố cơ bản của các phương pháp quasi-Newton là chúng thay thế ma trận Hessian trong khai triển Taylor (4.20) bởi $B(\theta)$. Cách thức cập nhật tham số mô hình cũng vì thế mà thay đổi:

$$\Delta^{(k)} = B^{-1}(\theta^{(k)})G(\theta^{(k)}) \tag{4.22}$$

Tại mỗi bước lặp, $B^{-1}(\theta)$ được cập nhật để phản ánh các thay đổi trong tham số tính từ bước lặp trước. Tuy nhiên, thay vì phải tính toán lại, $B^{-1}(\theta)$ chỉ cần phải cập nhật lại tại mỗi bước để phản ánh độ cong đo được trong bước lặp trước.

$$B(\theta^{(k)})^{-1}(G(\theta^{(k)}) - G(\theta^{(k-1)})) = \Delta^{k-1} \tag{4.23}$$

Việc xấp xỉ ma trận Hessian theo $B(\theta)$ cho phép phương pháp quasi-Newton hội tụ nhanh hơn so với phương pháp Newton truyền thống.

Phương pháp Limited memory quasi-Newton (L-BFGs) [11] cải tiến của phương pháp quasi-Newton để thực hiện tính tóan khi lượng bộ nhớ bị giới hạn. Những thực nghiệm gần đây cho thấy phương pháp Limited memory quasi-Newton vượt trội hơn hẳn so với các phương pháp khác bao gồm cả GIS, IIS, gradient liên hợp... trong việc tìm cực đại hàm log-likelihood.

4.3. Tổng kết chương

Chương này đề cập đến vấn đề ước lượng các tham số cho mô hình CRF bằng cách làm cực đại likelihood đồng thời giới thiệu một số phương pháp tìm cực đại của hàm log-likelihood như IIS, GIS, gradient liên hợp, quasi-Newton và L-BFGs nhằm phục vụ cho ước lượng tham số mô hình. Trong các phương pháp tìm cực trị hàm log-likelihood, phương pháp L-BFGs được đánh giá là vượt trội hơn hẳn so với các phương pháp khác.

Chương 5. Hệ thống nhận biết các loại thực thể trong tiếng Việt

Một hệ thống nhận biết loại thực thể trong tiếng Việt nếu ra đời sẽ góp phần quan trọng trong xử lý tiếng Việt và hiểu các văn bản tiếng Việt. Tuy rằng nhận biết loại thực thể là một bài toán cơ bản trong trích chọn thông tin và xử lý ngôn ngữ tự nhiên nhưng đối với tiếng Việt thì đây lại là một bài toán tương đối mới. Mặc dù có những khó khăn do đặc thù của tiếng Việt và tính chất tiên phong trong lĩnh vực nghiên cứu này, những thử nghiệm ban đầu của em cho tiếng Việt cũng đã đạt được những kết quả rất đáng khích lệ.

5.1. Môi trường thực nghiệm

5.1.1. Phần cứng

Máy Celeron III, chip 768 MHz, Ram 128 MB

5.1.2. 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,... 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). Hệ thống nhận biết loại thực thể cho tiếng Việt của em được xây dựng trên nền của Framework này.

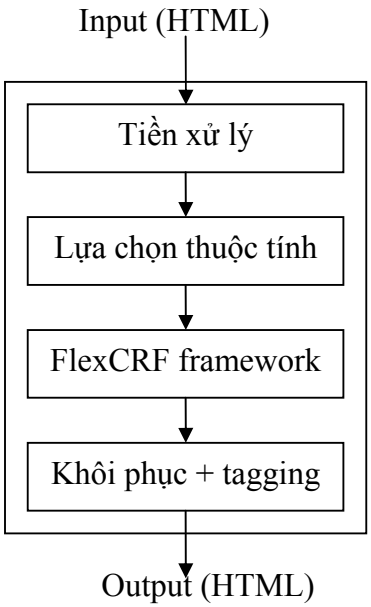
5.1.3. Dữ liệu thực nghiệm

Dữ liệu cho thực nghiệm gồm 50 bài báo lĩnh vực kinh doanh (khoảng gần 1400 câu) lấy từ nguồn http://www.vnexpress.net.

Dữ liệu ban đầu được cho qua bộ tiền xử lý để lọc bỏ các thẻ HTML và chuyển từ dạng mã hóa UTF-8 sang tiếng Việt không dấu mã hóa dạng Telex. Sau đó dữ liệu được gán nhãn bằng tay để phục vụ cho quá trình thực nghiệm.

5.2. Hệ thống nhận biết loại thực thể cho tiếng Việt

Các bước để gán nhãn cho một trang Web tiếng Việt được minh họa như hình vẽ dưới đây



Hình 8: Cấu trúc hệ thống nhận biết loại thực thể

5.3. Các tham số huấn luyện và đánh giá thực nghiệm

5.3.1. Các tham số huấn luyện

Một số tùy chọn trong FlexCRF framework cho quá trình huấn luyện:

Bảng 2: Các tham số trong quá trình huấn luyện

Tham số	Giá trị	Ý nghĩa
init_lamda_val	1.0	Giá trị khởi tạo cho các tham số trong mô hình
num_iterations	55	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	FlexCRF sử dụng phương pháp L-BFGs để ước lượng tham số mô hình. 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.

5.3.2. Đánh giá các hệ thống nhận biết loại thực thể

Các hệ thống nhận biết loại thực thể được đánh giá chất lượng thông qua ba độ đo: độ chính xác (precision), độ hồi tưởng (recall) và độ đo F (F-messure). Ba độ đo này được tính toán theo các công thức sau:

$$rec = \frac{correct}{correct + incorrect + mis\sin g} \quad (5.1)$$

$$pre = \frac{correct}{correct + incorrect + spurious} \quad (5.2)$$

$$F = \frac{2 * pre * rec}{pre + rec} \quad (5.3)$$

Ý nghĩa của các giá trị correct, incorrect, missing và spurious được định nghĩa như bảng sau:

Bảng 3: Các giá trị đánh gía một hệ thống nhận diện loại thực thể

Giá trị	Ý nghĩa
Correct	Số trường hợp được gán đúng.
Incorrect	Số trường hợp bị gán sai.
Missing	Số trường hợp bị thiếu
Spurious	Số trường hợp thừa

Một hệ thống nhận biết loại thực thể có thể được đánh gía ở mức độ nhãn hoặc ở mức độ cụm từ. Để hiểu rõ hơn vấn đề này chúng ta hãy xem xét ví dụ sau:

Ví dụ: giả sử hệ thống gán nhãn cụm từ "Phan Văn Khải" là "B_PER I_PER O". Ở mức độ nhãn, hệ thống gán đúng được 2 trong số 3 nhãn ví thế độ chính xác sẽ là 2/3. Ở mức độ cụm từ, ta muốn cả cụm này được đánh dấu là tên người hay chuỗi nhãn tương ứng phải là "B_PER I_PER I_PER", độ chính xác khi xét ở mức độ cụm từ sẽ là 0/1 (thực tế có một cụm tên thực thể nhưng hệ thống không đánh dấu đúng được cụm nào).

5.3.3. Phương pháp "10-fold cross validation"

Hệ thống thử nghiệm theo phương pháp "10-fold cross validation". Theo phương pháp này, dữ liệu thực nghiệm được chia thành 10 phần bằng nhau, lần lượt lấy 9 phần để huấn luyện và 1 phần còn lại để kiểm tra, kết quả sau 10 lần thực nghiệm được ghi lại và đánh giá tổng thể.

5.4. Lựa chọn các thuộc tính

Lựa chọn các thuộc tính từ tập dữ liệu huấn luyện là nhiệm vụ quan trọng nhất, giữ vai trò quyết định chất lượng của một hệ thống nhận biết loại thực thể. Các thuộc tính được lựa chọn càng tinh tế thì độ chính xác của hệ thống càng tăng. Do tiếng Việt thiếu các thông tin ngữ pháp (POS) cũng như các nguồn tài nguyên có thể tra cứu nên để có thể đạt được độ chính xác gần với độ chính xác đạt được với các hệ thống xây dựng cho tiếng Anh cần phải lựa chọn các thuộc tính một cách cẩn thận và hợp lý.

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.

Bước đầu thử nghiệm, em đưa ra một số mẫu vị từ ngữ cảnh sau:

5.4.1. Mẫu ngữ cảnh về từ vựng

Bảng 4: Các mẫu ngữ cảnh về từ vựng

Mẫu ngữ cảnh	Ý nghĩa
w:0,w:1	Dữ liệu quan sát được tại vị trí hiện tại và ngay sau vị trí hiện tại

Ví dụ: Áp dụng mẫu ngữ cảnh trên tại vị trí 1 trong chuỗi "3000 USD" ta được ngữ cảnh w:0:USD. Giả sử trong dữ liệu huấn luyện, từ USD trong chuỗi dữ liệu trên được gán nhãn I_CUR, kết hợp với ngữ cảnh ta có thể rút ra được một thuộc tính của chuỗi dữ liệu quan sát là

g_k = 1 nếu từ hiện tại là 'USD' và nhãn là I_CUR
0 nếu ngược lại

5.4.2. Mẫu ngữ cảnh thể hiện đặc điểm của từ

Bảng 5: Các mẫu ngữ cảnh thể hiện đặc điểm của từ

Mẫu ngữ cảnh	Ý nghĩa
initial_cap	Từ viết hoa chữ cái đầu tiên (có khả năng là thực thể)
all_cap	Từ gồm toàn các chữ cái viết hoa (có khả năng là ORG, ví dụ: EU, WTO...)
contain_percent_sign	Từ chứa kí tự % (có khả năng là thực thể PCT)
first_obsrv	Từ đầu tiên của câu (thông tin về viết hoa không có ý nghĩa)
uncaped_word	Từ viết thường (có khả năng không phải là thực thể)
valid_number	Từ hiện tại là một số hợp lệ, ví dụ: 123; 12.4

mark	Dấu câu như các dấu chấm, phẩy , hai chấm
4_digit_number	Nhiều khả năng là năm, ví dụ: năm 2005

5.4.3. Mẫu ngữ cảnh dạng regular expression

Bảng 6: Các mẫu ngữ cảnh dạng Regular Expression

Mẫu ngữ cảnh	Ví dụ	Ý nghĩa
^[0-9]+/[0-9]+/[0-9]+$	12/04/2005	Ngày tháng
^[0-9]+/[0-9]+$	22/5	Ngày tháng hoặc phân số
^[0-9][0-9][0-9][0-9]$	2005	Năm
^(T|t)hứ (hai|ba|tư|năm|sáu|bảy|)$ ^(C|c)hủ nhật$	Thứ hai	Ngày trong tuần
^[0-9]%$	7%	Phần trăm
^([0-9]|[A-Z])+$	3COM	Tên công ty

5.4.4. Mẫu ngữ cảnh dạng từ điển

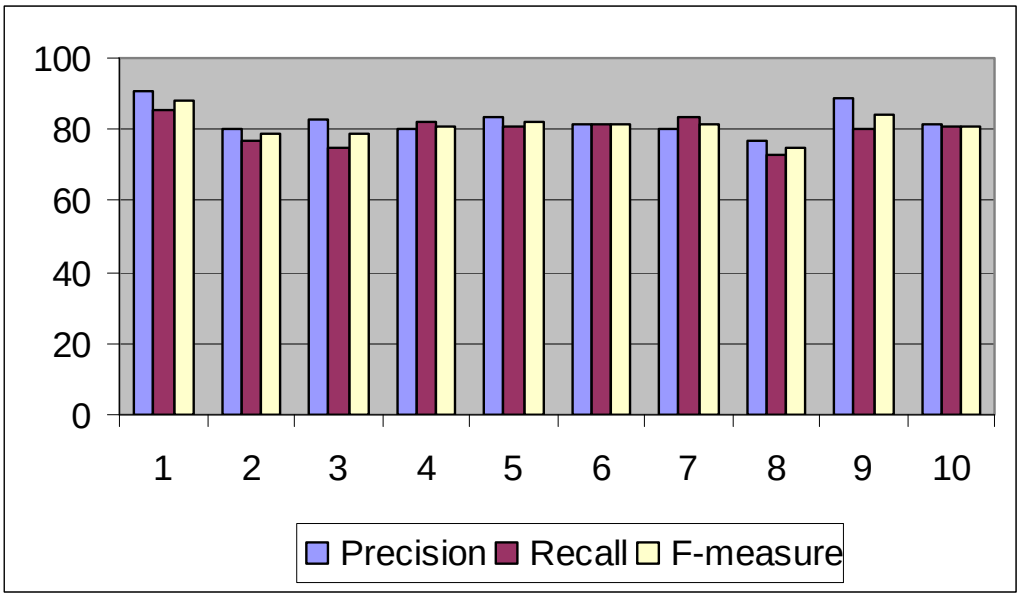
Các mẫu ngữ cảnh dạng này cho phép ta tra cứu trong một số danh sách cho trước. Các thông tin ngữ cảnh sinh ra từ các mẫu này rất có ích cho việc nhận biết loại thực thể. Nếu như trong tiếng Anh có các tài nguyên cho phép tra cứu như www.babyname.com (tra cứu các tên tiếng Anh) ... thì tiếng Việt hoàn toàn không có các nguồn tài nguyên như vậy, vì thế em phải thu thập và xây dựng các nguồn thông tin này từ đầu. Đây là một công việc rất mất thời gian nên em mới chỉ liệt kê thí điểm một vài trường hợp điển hình và vẫn chưa khai thác hết được thế mạnh của chúng.

Bảng 7: Các mẫu ngữ cảnh dạng từ điển

Mẫu ngữ cảnh	Ví dụ
first_name	Nguyễn, Trần, Lê ...
last_name	Hoa, Lan, Thắng ....
mid_name	Thị, Văn, Đình …
Verb	Sẽ, đã, phát biểu, nói ...
Time_marker	Sáng, trưa, chiều, tối
Loc_noun	Thị trấn, tỉnh, huyện, thủ đô, đảo, ...
Org_noun	Công ty, tổ chức, tổng công ty ...
Per_noun	Ông, bà, anh, chị, ...

5.5. Kết quả thực nghiệm

5.5.1. Kết quả của 10 lần thử nghiệm



Hình 9: Giá trị ba độ đo Precision, Recall, F-measure qua 10 lần thực nghiệm

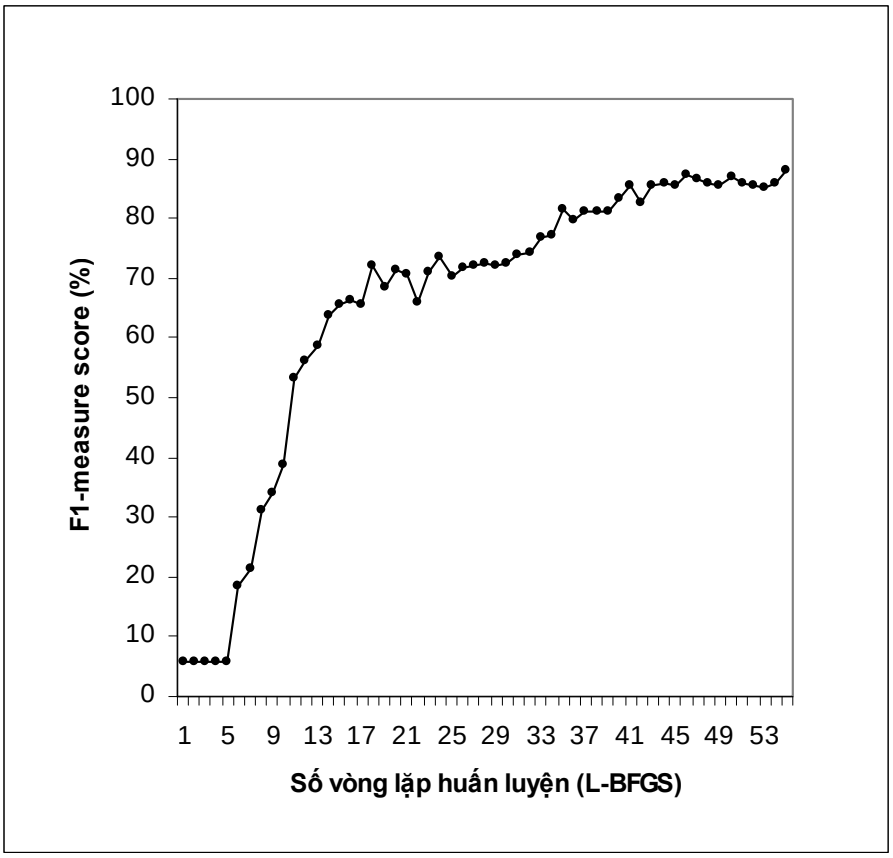
5.5.2. Lần thực nghiệm cho kết quả tốt nhất

Bảng 8: Đánh giá mức nhãn - Lần thực nghiệm cho kết quả tốt nhất

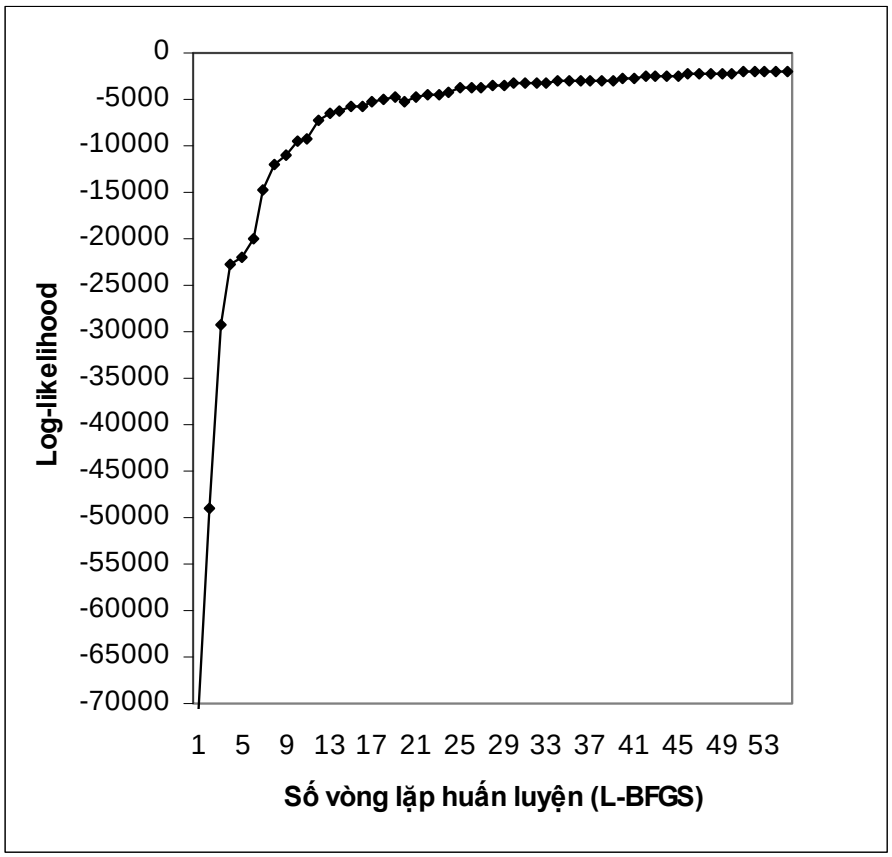
Label	Manual	Model	Match	Pre. (%)	Rec. (%)	F-Measure(%)
O	2132	2134	2101	98.4536	98.546	98.4998
B_LOC	91	97	83	85.567	91.2088	88.2979
I_LOC	55	59	51	86.4407	92.7273	89.4737
B_ORG	52	53	47	88.6792	90.3846	89.5238
B_TIME	58	67	54	80.597	93.1034	86.4
I_TIME	26	25	22	88	84.6154	86.2745
B_PER	13	13	12	92.3077	92.3077	92.3077
B_NUM	29	28	27	96.4286	93.1034	94.7368
I_NUM	3	2	2	100	66.6667	80
B_PCT	5	5	5	100	100	100
I_ORG	59	36	33	91.6667	55.9322	69.4737
B_CUR	12	12	11	91.6667	91.6667	91.6667
I_CUR	21	20	19	95	90.4762	92.6829
I_PER	15	18	15	83.3333	100	90.9091
B_MISC	10	7	5	71.4286	50	58.8235
I_MISC	4	3	3	100	75	85.7143
I_PCT	0	6	0	0	0	0
AVG1.				90.5981	85.3586	87.9003
AVG2.	2585	2585	2490	96.325	96.325	96.325

Bảng 9: Đánh giá mức cụm từ - Lần thực nghiệm cho kết quả tốt nhất

Chunk	Manual	Model	Match	Pre.(%)	Rec.(%)	F-Mesuare(%)
PER	13	13	12	92.31	92.31	92.31
LOC	91	97	82	84.54	90.11	87.23
ORG	52	53	40	75.47	76.92	76.19
PCT	5	5	5	100	100	100
MISC	10	7	5	71.43	50.00	58.82
NUM	29	28	27	96.43	93.10	94.74
TIME	58	67	54	80.60	93.10	86.40
CUR	12	12	11	91.67	91.67	91.67
ARG1.				86.55	85.90	86.23
ARG2.	270	282	236	83.69	87.41	85.51



Hình 10: Quá trình tăng F-measure qua các bước lặp



Hình 11: Quá trình tăng log-likelihood qua các bước lặp

5.5.3. Trung bình 10 lần thực nghiệm

Bảng 10: Đánh giá mức nhãn- Trung bình 10 lần thực nghiệm

Độ đo	Giá trị (%)
Precision	82.59756
Recall	79.89403
F-measure	81.18363

Bảng 11: Đánh giá ở mức "cụm từ" – trung bình 10 lần thực nghiệm

Độ đo	Giá trị (%)
Precision	81.855
Recall	79.351
F-measure	80.537

5.5.4. Nhận xét

Bước đầu thực nghiệm hệ thống nhận diện loại thực thể trong tiếng Việt cho kết quả tương đối khả quan. Tuy vẫn còn nhiều trường hợp nhập nhằng do những khó khăn đã đề cập trong chương 1 nhưng em tin rằng một khi đã xây dựng được tập dữ liệu huấn luyện đủ lớn, thu thập được các nguồn tra cứu dồi dào hơn và lựa chọn nhiều thuộc tính tốt hơn, hệ thống còn có thể đạt được độ chính xác cao hơn nữa trong tương lai.

Kết luận

Những vấn đề đã được giải quyết trong luận văn

Khóa luận đã hệ thống hóa một số vấn đề lý thuyết về trích chọn thông tin, bài toán nhận biết loại thực thể đồng thời trình bày, phân tích, đánh giá một số hướng tiếp cận bài toán nhận biết loại thực thể. Một số vấn đề và giải pháp đối với bài toán nhận biết loại thực thể cho tiếng Việt dựa trên mô hình CRF đã được đề xuất, thực nghiệm và thu được một số kết quả rất khả quan. Sau đây là một số nét chính mà luận văn đã tập trung giải quyết.

Chương một đưa ra một cái nhìn khái quát về trích chọn thông tin, bài toán nhận biết loại thực thể, mô hình hóa bài toán dưới dạng một bài tóan gán nhãn dữ liệu dạng chuỗi và những ứng dụng của bài tóan nhận diện loại thực thể từ đó thấy được sự cần thiết phải có một hệ thống nhận diện loại thực thể cho tiếng Việt.

Chương hai xem xét các hướng tiếp cận khác nhau để nhằm giải quyết bài toán nhận diện loại thực thể, đó là các phương pháp thủ công, phương pháp HMM, phương pháp MEMM. Chương này đi sâu vào phân tích đánh giá từng phương pháp, cho thấy sự thiếu linh hoạt của các phương pháp thủ công, sự nghèo nàn của các thuộc tính được chọn trong mô hình HMM và vấn đề "label bias" mà các mô hình MEMM gặp phải. Những đánh giá này lý giải vì sao em lại lựa chọn phương pháp học máy CRF là cơ sở để xây dựng hệ thống nhận diện loại thực thể cho tiếng Việt.

Chương ba đưa ra định nghĩa về CRF, giới thiệu nguyên lý cực đại hóa Entropy, thuật toán gán nhãn cho dữ liệu dạng chuỗi. Chương này cũng chứng minh rằng CRF là mô hình thích hợp nhất cho bài tóan nhận diện loại thực thể, cụ thể nó cho phép tích hợp các thuộc tính phong phú đa dạng của chuỗi dữ liệu quan sát, bản chất phân phối toàn cục giúp cho các mô hình CRF tránh được vấn đề "label bias" mà MEMM gặp phải.

Chương bốn hệ thống các phương pháp ước lượng các tham số cho các mô hình CRF, đó là các phương pháp lặp (IIS, GIS), các phương pháp dựa trên vector gradient như gradient liên hợp, quasi-Newton, L-BFGs. Trong số các phương pháp này, L-BFGs được đánh giá tốt nhất, đây cũng chính là phương pháp mà FlexCRFs – một CRF framework - sử dụng để ước lượng tham số cho mô hình.

Chương năm trình bày hệ thống nhận diện loại thực thể cho tiếng Việt và đề xuất các phương pháp lựa chọn thuộc tính cho việc nhận diện các loại thực thể trong các văn bản tiếng Việt. Chương này cũng đưa ra các kết quả của hệ thống nhận diện loại thực thể tiếng Việt qua một số lần thực nghiệm.

Công việc nghiên cứu trong tương lai

Mặc dù kết quả phân loại thực thể của hệ thống có thể tốt hơn nữa nhưng do thời gian có hạn nên em mới chỉ dừng lại ở con số trung bình là 80%, trong thời gian tới, em sẽ tiếp tục nghiên cứu nhằm cải thiện hệ thống, em tin rằng kết quả này có thể tăng lên xấp xỉ 90% ở mức cụm từ.

Trên cơ sở hệ thống nhận diện loại thực thể tiếng Việt hiện nay, em dự định sẽ mở rộng và cụ thể hóa các loại thực thể như phân nhỏ loại thực thể chỉ địa danh thành các loại thực thể chỉ đất nước, sông ngòi, ...

Tìm hiểu và xây dựng một hệ thống nhận diện mối quan hệ giữa các thực thể như tìm ra mối quan hệ như nơi sinh của một người, về chức vụ một người trong một công ty tổ chức ...

Xây dựng một ontology chỉ địa danh, tổ chức, ... cho tiếng Việt. Tích hợp ontology và hệ thống nhận diện loại thực thể vào máy tìm kiếm tiếng Việt Vinahoo nhằm phục vụ việc tìm kiếm hướng thực thể.

Phụ lục: Output của hệ thống nhận diện loại thực thể tiếng Việt

Bảng Chú thích:

Màu	Loại thực thể	Ý nghĩa
Nâu	LOC	Tên địa danh
Tía	ORG	Tên tổ chức
Xanh nước biển	PER	Tên người
Đỏ	PCT	Phần trăm
Xanh lá cây	TIME	Ngày tháng, thời gian
Tím	CUR	Tiền tệ
Xanh nhạt	NUM	Số
Da cam	MISC	Những loại thực thể khác

Kết quả sau khi hệ thống gán nhãn một số chuỗi dữ liệu quan sát

Thứ năm,16/12/2004,15:11 GMT+7.

Cao Xumin , Chủ tịch Phòng Thương mại Xuất Nhập khẩu thực phẩm của Trung Quốc , cho rằng , cách xem xét của DOC khi đem so sánh giá tôm của Trung Quốc với giá tôm của Ấn Độ là vi phạm luật thương mại .

Để đảm bảo lợi ích của Nhà nước và doanh nghiệp, sau thời điểm bàn giao tài sản , VMS mới có thể tiến hành kiểm kê và thuê tổ chức tư vấn xác định giá trị doanh nghiệp .

EU thúc đẩy quan hệ thương mại với Trung Quốc (24/02).

Hiệp hội chất lượng Thượng Hải đã phỏng vấn 2.714 khách hàng ở 29 siêu thị quanh thành phố trong tháng qua.

Thủ tướng Trung Quốc Ôn Gia Bảo vừa cho biết , năm nay nước này sẽ giảm tốc độ tăng trưởng kinh tế xuống còn 8% so với con số 9,4% trong năm 2004 nhằm đạt được sự phát triển ổn định hơn .

Hãng cũng sẽ mở rộng mạng lưới của mình sang Australia và Canada. OPEC giữ nguyên sản lượng khai thác dầu.

Theo kế hoạch , vòng 2 của cuộc thi lần này với 6 đội chơi sẽ tổ chức đồng thời ở Hong Kong , TP HCM và Australia .

' Đại diện thương mại EU không nên lãnh đạo WTO ' (12/03) .
VN miễn thị thực cho công dân 4 nước Bắc Âu (20/04) .
Giá dầu thế giới giảm nhẹ sau tuyên bố của OPEC (25/02) .
TP HCM tổ chức ngày hội du lịch nhân dịp 30/4 (21/04) .
Trước thực trạng này , những du khách đến lễ hội mà không đặt phòng trước chỉ còn cách thuê các khách sạn ở phía ngoài , cách xa trung tâm thành phố .
Khi gia nhập WTO , môi trường đầu tư của Trung Quốc cả về " môi trường cứng " (cơ sở hạ tầng) lẫn " môi trường mềm " (cơ chế chính sách) sẽ được cải thiện hơn nữa , Trung Quốc sẽ trở thành một trong những "điểm nóng " thu hút đầu tư nước ngoài của thế giới .
- Cụ thể chúng ta sẽ làm gì để đẩy nhanh tiến độ gia nhập WTO?
Nhật đã khuyến cáo công dân của họ ở Trung Quốc chú ý đến an ninh khi làm sóng biểu tình bắt đầu cách đây vài tuần.
Nỗ lực của Trung Quốc gia nhập WTO (28/12) .
" Có rất nhiều thanh niên Nhật hiểu biết về Trung Quốc " .
Trung Quốc mở màn cuộc chiến thép mới (14/01) .
Thêm 2 công ty đấu giá cổ phần qua sàn Hà Nội (12/04) .
Khối lượng giao dịch không có biến động lớn so với tuần trước khiến thị trường vẫn ở thế nằm ngang .
Sự nóng bỏng của thị trường vàng đen trong những ngày qua khiến giới phân tích đưa ra nhận định , thị trường nhiên liệu ngày càng nhạy cảm với những nhân tố vĩ mô như chính sách của Tổ chức các nước xuất khẩu dầu mỏ (OPEC) , nhu cầu sử dụng của những người khổng lồ như Mỹ , Trung Quốc và Ấn Độ .
Dầu thô chỉ còn 50 USD /thùng (14/04).
Hồi tháng 12 năm ngoái , Tổng thống Mỹ George Bush , người tháo ngòi cuộc chiến tranh thép với EU và một số nước châu Á , cũng đã phải dỡ bỏ thuế suất cao sau nhiều lần WTO đưa ra lời cảnh cáo .
Bước dài từ CEPT đến WTO (04/01) .
Lộ trình chuẩn bị gia nhập WTO của Việt Nam (22/12) .
Trên thực tế , Chính phủ Trung Quốc đã đổ nhiều tiền của cho ngành thép trong nước , đồng thời không quên cảnh báo bằng mọi cách sẽ lấn át các đối thủ khác , ít nhất là trong vòng 10 năm tới .
Về lâu dài, từ nay cho đến tháng 3 sang năm, doanh thu của toàn Thai Airways sẽ giảm khoảng 2-3% do Phuket là một trong nhưng thị trường chính.

Ngay sau khi thảm họa xảy ra , sân bay Phuket đã đống cửa vài giờ và đã hoạt động lại sau 6 giờ.

Tính đến hôm qua , 60% khách du lịch nươc ngoài đã hủy chỗ ở khách sạn và khu nghỉ dưỡng ở Phuket .

Tài liệu tham khảo

[1]. A.Berger, A.D.Pietra, and J.D.Pietra.A maximum entropy approach to natural langauge processing. *Computational Linguistics*, 22(1):39-71, 1996.

[2]. Adam Berger. The Improved Iterative Scaling Algorithm: A gentle Introdution. School of Computer Science, Carnegie Mellon University

[3]. Andrew Borthwick. A maximum entropy approach to Named Entity Recognition. New York University, 1999

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

[5]. A.McCallum, D.Freitag, and F. Pereira. Maximum entropy markov models for information extraction and segmentation. In Proc. Iternational Conference on Mechine Learning, 2000, pages 591-598.

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

[7]. Andrew Moore. Hidden Markov Models Tutorial Slides.

[8]. A.Ratnaparkhi.A maximum entropy model for part-of-speech tagging.In Proc. Emparical Methods for Natural Language Processing, 1996.

[9]. Basilis Gidas. Stochastic Graphical Models and Applications, 2000. University of Minnesota.

[10]. David Barber. An Introduction to Graphical Models.

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

[12]. F.Sha and F.Pereira.Shallow parsing with conditional random fields. In Proc. Human Language Technology/ the Association for Computational Linguistics North American Chapter, 2003.

[13]. GuoDong Zhou, Jian Su. Named Entity Recognition using an HMM-based Chunk Tagger.

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

[15]. Hanna Wallach. Efficient Training of Conditional Random Fields. University Of Edinburgh, 2002

[16]. Hieu Phan, Minh Nguyen, Bao Ho – Japan Advanced Institute of Science and Technology,Japan , and Susumu Horiguchi- Tokosu University, Japan. Improving Discriminative Sequential Learning with Rare-but-Important Associations. SIGKDD ’05 Chicago, II, USA, 2005.

[17]. J.Lafferty, A.McCallum, and F.Pereira.Conditional random fields: probabilistic models for segmenting and labeling sequence data. In Proc. ICML, 2001.

[18]. John Lafferty, Yan Liu, Xiaojin Zhu, School of Computer Science – Carnegie Mellon University, Pittsburgh, PA 15213. Kernel Conditonal Random Fields: Representation, Clique Selection and Semi-Supervised Learning. CMS-CS-04-115, February 5, 2004.

[19]. Rabiner.A tutorial on hidden markov models and selected applications in speech recognition. In Proc. the IEEE, 77(2):257-286, 1989.

[20]. Robert Malouf, Alfa-Informatica Rijksuniversiteit Groningen, Postbus 716 9700AS Groningen The Newtherlands. A comparison of Algorithms for maximum entropy parameter estimation.

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

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

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

[24]. Xuming He, Richard S. Zemel, Miguel Á. Carreira-Perpinan, Department of Computer Science, University of Toronto. Multiscale Conditional Random Fields for Image Labeling.

[25]. Yasemin Altun and Thomas Hofmann, Department of Computer Science, Brown University, Providence, RI. Large Margin Methods for Label Sequence Learning.

[26]. Yasemin Altun, Alex J. Smola, Thomas Hofmann. Exponential Faminlies for Conditional Random Fields.

[27]. Walter F.Mascarenhas. The BFGS method with exact line searches fails for non-convex objective functions. Published May 7, 2003

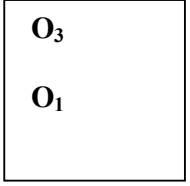
[28]. Web site: http://web.mit.edu/wwmatch . Optimization

[29]. Web site: http://www.mtm.ufsc.br/ . Shannon Entropy

[30]. Web site: http://www.cs.nyu.edu/cs/faculty/grishman/muc6.html . Information about the sixth Message Understanding Conference.

[31]. Web site: http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/muc_7_toc.html . Information about the seventh Message Understanding Conference.

[32]. William W.Cohen, Adrew McCallum. Slides "Information Extraction from the World Wide Web", KDD 2003.



[1]. Andrew Borthwick. A maximum entropy approach to Named Entity Recognition. Doctor of Philosophy, New York University, September 1999

[2]. A.McCallum, D.Freitag, F. Pereira. Maximum entropy markov models for information extraction and segmentation. In Proc. ICML 2000, pages 591-598.

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

[4]. GuoDong Zhou, Jian Su. Named Entity Recognition using an HMM-based Chunk Tagger. ACL Philadenphia, July 2002, pp. 473-480

[5]. Hanna Wallach. Efficient Training of Conditional Random Fields. Doctor of Philosophy, University Of Edinburgh, 2002

[6]. Hieu Phan, Minh Nguyen, Bao Ho, and Susumu Horiguchi. Improving Discriminative Sequential Learning with Rare-but-Important Associations. ACM SIGKDD Chicago, IL, USA, August 21-24, 2005 (to appear).

[7]. J.Lafferty, A.McCallum, and F.Pereira.Conditional random fields: probabilistic models for segmenting and labeling sequence data. In Proc. ICML , pages 282-290,2001

[8]. Rabiner.A tutorial on hidden markov models and selected applications in speech recognition. In Proc. the IEEE, 77(2):257-286, 1989.

[9]. William W.Cohen, Adrew McCallum. Slides “Information Extraction from the World Wide Web”, KDD 2003

[10]. P.X.Hieu, N.L.Minh. http://www.jaist.ac.jp/~hieuxuan/flexcrfs/flexcrfs.html

[11]. Website: http://www.itl.nist.gov/iaui/894.02/related_projects/muc/index.html