ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Vũ Đăng Tùng

TỐI ƯU BẢNG CỤM TỪ TRONG DỊCH MÁY THỐNG KÊ DỰA VÀO CỤM

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành : Công nghệ thông tin

HÀ NỘI - 2013

ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

Vũ Đăng Tùng

TỐI ƯU BẢNG CỤM TỪ TRONG DỊCH MÁY THỐNG KÊ DỰA VÀO CỤM

KHÓA LUẬN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành : Công nghệ thông tin

Cán bộ hướng dẫn: Ts.Nguyễn Văn Vinh

HÀ NỘI – 2013

TỐI ƯU HÓA BẢNG CỤM TỪ TRONG DỊCH MÁY THỐNG KÊ DỰA VÀO CỤM

Vũ Đăng Tùng

Khóa QH-2009-I/CQ ngành Công Nghệ Thông Tin

Tóm tắt khóa luận tốt nghiệp :

Ngôn ngữ tự nhiên(NLP) là ngôn ngữ chúng ta giao tiếp hằng ngày .Mặc dù con người có thể dễ dàng hiểu được ngôn ngữ nhưng đối với máy tính và các hệ thống thì việc hiểu được dữ liệu thô truyền vào quả là một điều không hề đơn giản.

Dịch máy thống kê(SMT) là mô hình dịch tự động từ ngôn ngữ này sang một ngôn ngữ khác dựa trên mô hình sinh ra một cách tự động từ ngữ liệu song ngữ.

Việc nghiên cứu dịch máy thống kê gặp phải rất nhiều khó khăn và thách thức cần được giải quyết.

Trong dịch máy thống kê thì dịch máy thống kê dựa vào cụm là một phương pháp hay được sử dụng.Nó có nhiều ưu điểm hơn phương pháp dịch dựa theo từ về chất lượng và thời gian dịch.

Tuy nhiên nó cũng gặp nhiều vấn đề cần được giải quyết.Trong luận văn này sẽ trình bày khái quát một số phương pháp tối ưu hóa cụm từ dựa vào cụm.Trong luận văn có sử dụng hệ thống MOSES và các công cụ mã nguồn mở SRILM trong dịch máy.

Từ khóa: SMT,NLP

LỜI CẢM ƠN

Đầu tiên, cho phép tôi gửi lời cảm ơn sâu sắc tới Ts.Nguyễn Văn Vinh, người đã trực tiếp hướng dẫn, chỉ bảo và tạo điều kiện cho tôi trong quá trình hoàn thành luận văn này.

Đồng thời tôi cũng xin gửi lời cảm ơn chân thành tới các thầy cô giáo trường Đại học Công Nghệ-ĐHQGHN, đặc biệt là các thầy cô trong bộ môn Khoa học Máy tính , những người đã trực tiếp giảng dạy và cho tôi những kiến thức cơ bản về xử lý ngôn ngữ tự nhiên trong bộ môn cùng tên,đã hướng dẫn và tạo điều kiện cho tôi trong quá trình học tập và thực hành ở trường.

Cuối cùng, tôi xin gửi lời cảm ơn tới tất cả các bạn đồng học và gia đình đã ủng hộ, giúp đỡ tôi hoàn thành luận văn này .

Mục Lục

THUẬT NGỮ

SMT	Dịch máy thống kê
NLP	Ngôn ngữ tự nhiên
PB	Bảng cụm từ
S9	Thuật toán Simple-9
PB-SMT	Bảng cụm từ trong dịch máy thống kê
PR-Enc	Mã hóa thứ hạng cụm
SB	Cụm từ con (Subphrase)

CHƯƠNG I: TỔNG QUAN VỀ DỊCH MÁY

Phần này nêu lên các khái niệm cơ bản về dịch máy và thống kê.

1.1-Ngôn ngữ tự nhiên

Ngôn ngữ tự nhiên là những ngôn ngữ được con người sử dụng trong các giao tiếp hàng ngày: nghe, nói, đọc, viết . Mặc dù con người có thể dễ dàng hiểu và học các ngôn ngữ tự nhiên; việc làm cho máy hiểu được ngôn ngữ tự nhiên không phải là chuyện dễ dàng. Sở dĩ có khó khăn là do ngôn ngữ tự nhiên có các bộ luật, cấu trúc ngữ pháp phong phú hơn nhiều các ngôn ngữ máy tính, hơn nữa để hiểu đúng nội dung các giao tiếp, văn bản trong ngôn ngữ tự nhiên cần phải nắm được ngữ cảnh của nội dung đó. Do vậy, để có thể xây dựng được một bộ ngữ pháp, từ vựng hoàn chỉnh, chính xác để máy có thể hiểu ngôn ngữ tự nhiên là một việc rất tốn công sức và đòi hỏi người thực hiện phải có hiểu biết sâu về ngôn ngữ học. Do đó cần phải tìm ra một phương pháp dịch tư động tối ưu để làm giảm công sức trong vấn đề về dịch ngôn ngữ nói chung.

1.2- Mô hình ngôn ngữ

Phương pháp dịch máy thông kê dựa trên xác xuất để xuất hiện ngôn ngữ đích khi cho đầu vào là một ngôn ngữ nguồn . Việc thống kê dựa trên bộ ngữ liệu có sẵn..Và ta chỉ xác định xác xuất nào là lớn nhất để chọn ra kết quả ngôn ngữ đích phù hợp .

Ví dụ : Khi áp dụng mô hình ngôn ngữ cho tiếng Việt

P[“hôm nay là thứ hai]=0.003

P[“hai nay là thứ hôm”]=0

Mô hình ngôn ngữ được áp dụng trong nhiều lĩnh vực của xử lý ngôn ngữ tự nhiên..Có nhiều hướng tiếp cận mô hình ngôn ngữ nhưng chủ yếu được xây dựng theo mô hình ngôn ngữ N-gram

Mô hình ngôn ngữ N-gram:

*Nhiệm vụ của mô hình ngôn ngữ là cho biết xác suất của một câu w1w2...wm là bao nhiêu. Theo công thức Bayes: P(AB) = P(B|A) * P(A), thì:*

*P(w1w2…wm) = P(w1) * P(w2|w1) * P(w3|w1w2) *…* P(wm|w1w2…wm-1)*

Theo công thức này, mô hình ngôn ngữ cần phải có một lượng bộ nhớ vô cùng lớn để có thể lưu hết xác suất của tất cả các chuỗi độ dài nhỏ hơn m. Rõ ràng, điều này là không thể khi m là độ dài của các văn bản ngôn ngữ tự nhiên (m có thể tiến tới vô cùng). Để có thể tính được xác suất của văn bản với lượng bộ nhớ chấp nhận được, ta sử dụng xấp xỉ Markov bậc n:

P(wm|w1,w2,…, wm-1) = P(wm|wm-n,wn-m+1, …,wm-1)

Nếu áp dụng xấp xỉ Markov, xác suất xuất hiện của một từ (wm) được coi như chỉ phụ thuộc vào n từ đứng liền trước nó (wm-nwm-n+1…wm-1) chứ không phải phụ thuộc vào toàn bộ dãy từ đứng trước (w1w2…wm-1). Như vậy, công thức tính xác suất văn bản được tính lại theo công thức:

*P(w1w2…wm) = P(w1) * P(w2|w1) * P(w3|w1w2) *…* P(wm-1|wm-n-1wm-n …wm-2)* P(wm|wm-nwm-n+1…wm-1)*

Với công thức này, ta có thể xây dựng mô hình ngôn ngữ dựa trên việc thống kê các cụm có ít hơn n+1 từ. Mô hình ngôn ngữ này gọi là mô hình ngôn ngữ N-gram.

Một cụm N-gram là 1 dãy con gồm n phần tử liên tiếp nhau của 1 dãy các phần tử cho trước.

Thí dụ với mô hình 3-Gram:

v = "Tôi đang đọc sách" (I=4)

p(Tôi đang đọc sách) = p(Tôi | <bắt-đầu-câu><bắt-đầu-câu>) *

p(đang | <bắt-đầu-câu>Tôi) *p(đọc | tôi đang) *p(sách | đang đọc)

Tính p(z|xy): P(đọc|tôi đang)= c(tôi đang đọc)/ c(tôi đang)

1.3- Dịch máy

Dịch tự động hay còn gọi là dịch máy là một trong những ứng dụng quan trọng của xử lý ngôn ngữ tự nhiên,là sự kết hợp của ngôn ngữ ,dịch thuật và khoa học máy tính.Như tên gọi dịch tự động là việc thực hiện dịch một ngôn ngữ đầu vào (ngôn ngữ này gọi là ngôn ngữ nguồn) sang một hoặc nhiều ngôn ngữ khác (gọi là ngôn ngữ đích) bằng các công cụ ,phần mềm trên máy tính đã được lập trình sẵn mà không cần có sự can thiệp của con người .

Do được lập trình sẵn bằng công cụ,thuật toán trên máy tính nên hầu hết việc dịch tự động đều mang tính sát nghĩa,hoặc mang tính tương đối..

Ngày nay người ta đã phát triển nhiều phương pháp để tối ưu hóa khả năng dịch của máy tính

Dịch máy có hai hướng tiếp cận chính đó là :

- Hướng luật(Rules-based): dịch dựa vào các luật viết tay.Các luật này dựa trên từ vựng hoặc cú pháp của ngôn ngữ.Ưu điểm của phương pháp này là có thể giải quyết được một số trường hợp dịch nhưng lại mất nhiều công sức và tính khả chuyển không cao.
- Thống kê (Statistical) : tạo ra bản sử dụng phương pháp thống kê dựa trên bản dịch song ngữ.

1.4 -Dịch máy thống kê (SMT) :

Dịch máy thống kê : là một phương pháp dịch máy trong đó các bản dịch được tạo ra trên cơ sở các mô hình thống kê có các tham số được bắt nguồn từ việc phân tích các cặp câu song ngữ. Các phương pháp tiếp cận thống kê tương phản với các phương pháp tiếp cận dựa trên luật trong dịch máy cũng như với dịch máy dựa trên ví dụ.

Thay vì xây dựng các từ điển ,các quy luật chuyển đổi bằng tay,hệ dịch này tự động xây dựng các từ điển ,các quy luật dựa trên kết quả thống kê có được từ kho ngữ liệu.Chính vì vậy dịch máy thống kê có tính khả chuyển cao và áp dụng được cho bất cứ cặp ngôn ngữ nào .

Ý tưởng đầu tiên của dịch máy thống kê đã được giới thiệu bởi Warren Weaver vào năm 1949 , bao gồm cả những ý tưởng của việc áp dụng lý thuyết thông tin của Claude Shannon. Dịch máy thống kê được tái giới thiệu vào năm 1991 bởi các nhà nghiên cứu làm việc tại Trung tâm nghiên cứu Thomas J. Watson của IBM và đã góp phần đáng kể trong sự hồi sinh việc quan tâm đến dịch máy trong những năm gần đây. Ngày nay nó là phương pháp dịch máy được nghiên cứu nhiều nhất.

1.4.1 Cơ sở của phương pháp dịch máy thống kê.

Ý tưởng của dịch máy thống kê đến từ lý thuyết thông tin.Tài liệu được dịch phân bố theo xác suất $p(e|f)$ trong đó e là ngôn ngữ đích(Ví dụ : Tiếng việt) dịch từ f là ngôn ngữ nguồn (ví dụ : Tiếng Anh).

Việc tính toán mô hình xác suất $p(e|f)$ thường được tiếp cận một cách trực quan qua quy tắc Bayes

Đó là $P(e|f)= \frac{\boldsymbol{P(e).P(f|e)}}{\boldsymbol{P(f)}}$

Trong đó : P(f|e) là xác suất để chuỗi nguồn (f) là bản dịch của chuỗi đích e (xác xuất này gọi là mô hình dịch) và P(e) là xác suất chuỗi e thực sự xuất hiện trong ngôn ngữ đích, xác suất này gọi là mô hình ngôn ngữ. Phân tích này giúp tách các vấn đề thành hai bài toán con. Bản dịch tốt nhất $\tilde{e}$ được tìm bằng cách chọn ra bản có xác suất cao nhất:

$$\tilde{e} = arg\max_{e \in e^*} p(e|f) = arg\max_{e \in e^*} p(f|e)p(e)$$

Do hệ thống dịch không thể lưu trữ tất cả các chuỗi nguồn và bản dịch của chúng, một tài liệu thường được dịch từng câu một, nhưng ngay cả việc lưu tất cả câu cũng không khả thi. Mô hình ngôn ngữ thường được tính xấp xỉ bằng mô hình n-gram, và cách tiếp cận tương tự đã được áp dụng cho mô hình dịch, nhưng có thêm sự phức tạp do độ dài câu và thứ tự từ khác nhau trong các ngôn ngữ.

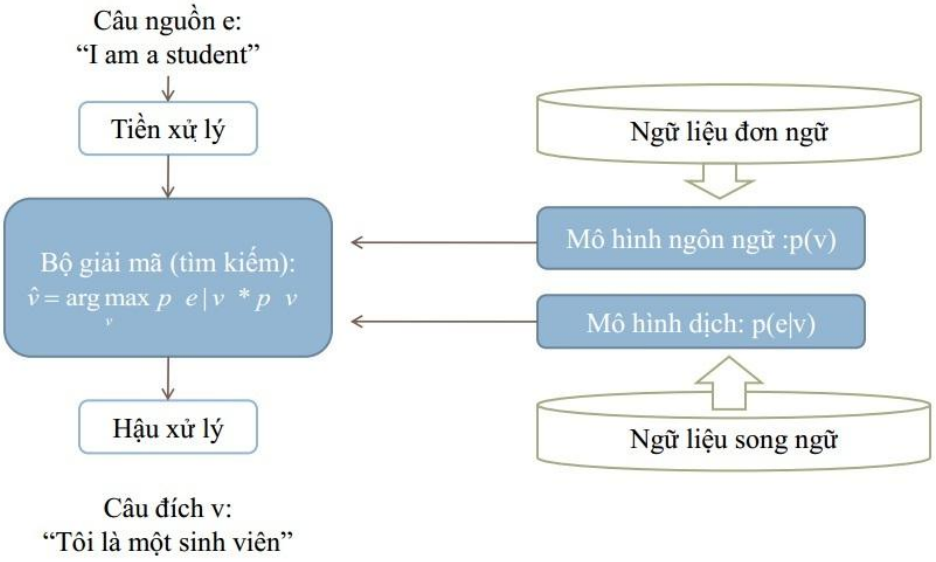
Các mô hình dịch thống kê ban đầu thường dùng mô hình lấy cơ sở theo từ (mô hình 1-5 mô hình Markov ẩn của IBM của Stephan Vogel và Mô hình 6 của Franz-Joseph Och), nhưng những tiến bộ đáng kể đã được thực hiện từ khi có mô hình lấy cơ sở theo cụm từ. Các công trình nghiên cứu gần đây đã kết hợp cú pháp hoặc cấu trúc bán-cú pháp để làm tăng chất lượng dịch.

Để mô hình dịch là chính xác ,thì công việc tiếp theo là tìm ra tất cả các câu e* có thể có trong ngôn ngữ đích.Đó chính là nhiệm vụ của bộ giải mã(Decoder) .

Một bộ giải mã gồm 3 thành phần :

- Mô hình ngôn ngữ : Tính toán được xác suất của ngôn ngữ nguồn
- Mô hình dịch : Cho biết xác suất của câu ngôn ngữ nguồn là bản dịch từ câu ngôn ngữ đích
- Bộ giải mã: Tìm kiếm tất cả các câu ngôn ngữ đích e có thể có từ câu ngôn ngữ nguồn f.

Ví dụ : Mô hình dịch từ Tiếng Anh -> Tiếng Việt .



1.4.2 Dịch máy thống kê trên cơ sở từ

Các đơn vị cơ bản của bản dịch là một từ trong ngôn ngữ tự nhiên. Một ví dụ về một hệ thống dịch máy thống kê trên cơ sở từ là phần mềm tự do Giza++ (giấy phép GPL), dùng để tập huấn cho các mô hình dich IBM, mô hình HMM.

Dịch máy thống kê trên cơ sở từ không sử dụng rộng rãi ngày nay, thay vào đó là dịch máy thống kê trên cơ sở cụm từ. Hầu hết các hệ thống dựa trên cụm từ vẫn còn sử dụng Giza++ để gióng hàng câu, trích rút ra các cặp câu song ngữ và mô hình ngôn ngữ. Vì những ưu thế của Giza++, hiện nay có một số nỗ lực đưa áp dụng tính toán phân tán trực tuyến cho phần mềm này

1.4.3 Dịch máy thống kê dựa trên cơ sở cụm từ.

Dịch máy thống kê trên cơ sở cụm từ có mục đích là để giảm bớt các hạn chế của dịch máy thống kê trên cơ sở từ bằng cách dịch cụm từ, trong đó độ dài cụm từ nguồn và cụm từ đích có thể khác nhau. Các cụm từ trong kỹ thuật này

thường không cụm từ theo nghĩa ngôn ngữ học mà là các cụm từ được tìm thấy bằng cách sử dụng phương pháp thống kê để trích rút từ các cặp câu. Việc sử dụng các cụm từ theo nghĩa ngôn ngữ học (tức là dựa trên cú pháp, xem phân loại cú pháp) làm giảm chất lượng của dịch máy bằng phương pháp này.

1.4.4 Dịch máy thống kê trên cơ sở cú pháp

Dịch máy thống kê trên cơ sở cú pháp dựa trên ý tưởng của dịch các đơn vị cú pháp (phân tích cây của câu), hơn là những từ đơn hay cụm từ (như trong dịch máy thống kê trên cơ sở cụm từ). Ý tưởng này đã xuất hiện từ lâu, tuy nhiên phiên bản thống kê của ý tưởng này chỉ được hình thành khi có những bộ phân tích ngẫu nhiên mạnh mẽ trong những năm 1990.

Để áp dụng phương pháp này một cách đầy đủ, cần thực hiện việc tìm kiếm trên tất cả các chuỗi e^* của ngôn ngữ đích. Khối lượng tìm kiếm này rất lớn, và nhiệm vụ thực hiện tìm kiếm hiệu quả là công việc của một bộ giải mã dịch máy, sử dụng nhiều kỹ thuật để hạn chế không gian tìm kiếm nhưng vẫn giữ chất lượng dịch thuật chấp nhận được. Kỹ thuật đánh đổi giữa chất lượng và thời gian tính toán cũng có thể được tìm thấy trong nhận dạng tiếng nói.

Do hệ thống dịch không thể lưu trữ tất cả các chuỗi nguồn và bản dịch của chúng, một tài liệu thường được dịch từng câu một, nhưng ngay cả việc lưu tất cả câu cũng không khả thi. Mô hình ngôn ngữ thường được tính xấp xỉ bằng mô hình n-gram, và cách tiếp cận tương tự đã được áp dụng cho mô hình dịch, nhưng có thêm sự phức tạp do độ dài câu và thứ tự từ khác nhau trong các ngôn ngữ.

Các mô hình dịch thống kê ban đầu thường dùng mô hình lấy cơ sở theo từ (mô hình Markov ẩn của IBM của Stephan Vogel và Mô hình Franz-Joseph Och), nhưng những tiến bộ đáng kể đã được thực hiện từ khi có mô hình lấy cơ sở theo cụm từ. Các công trình nghiên cứu gần đây đã kết hợp cú pháp hoặc cấu trúc bán-cú pháp để làm tăng chất lượng dịch

1.5- Một số vấn đề gặp phải trong dịch máy

1.5.1 Vấn đề gióng hàng câu

Trong khi phương pháp dịch máy thống kê dựa trên những cặp câu song ngữ, thì một câu trong ngôn ngữ này có thể được dịch ra nhiều câu khác nhau trong ngôn

ngữ khác và ngược lại. Việc gióng hàng câu có thể được thực hiện thông qua các thuật toán gióng hàng Gale-Church.

1.5.2 Từ ghép

Một vấn đề nữa trong dịch máy thống kê là xác định các từ ghép,hoặc cặp từ ghép..

Một số ngôn ngữ rất khó xác định từ ghép trong câu,hoặc có trường hợp có nhiều phương án để xác định từ ghép..Chính vì vậy cũng gây khó khan trong việc xác định đầu vào của ngôn ngữ nguồn..

1.5.3 Thành ngữ

Tùy thuộc vào bộ cặp câu sử dụng, các thành ngữ có thể không được dịch thoát nghĩa hay theo nghĩa bóng, ẩn nghĩa của chúng.Trong khi đó số lượng thành ngữ của mỗi ngôn ngữ lại rất phong phú..Chúng ta không thể đặt ra các ngoại lệ với các thành ngữ lại trong lập trình dịch máy được..Điều này gây khó khăn cho kết quả dịch máy

1.5.4 Hình thái học

1.5.5 Khác biệt trong thứ tự từ

Thứ tự từ trong các ngôn ngữ là khác nhau. Một số ngôn ngữ có thể được phân loại bằng cách đặt tên theo thứ tự điển hình của chủ ngữ (S), động từ (V) và đối tượng (O) trong một câu và có thể có các ngôn ngữ theo dạng, chẳng hạn, SVO hoặc VSO. Ngoài ra còn có thêm sự khác biệt trong thứ tự từ, ví dụ, khi có những yếu tố ngữ pháp phụ trợ, ví dụ thứ tự từ của câu hỏi khác câu khẳng định.

Để giải quyết vấn đề sắp xếp thứ tự từ, nhiều bản dịch ứng với các thứ tự từ khác nhau có thể được sinh ra, sau đó các bản dịch này được xếp hạng về xác suất xuất hiện, với sự giúp đỡ của mô hình ngôn ngữ, và bản dịch có xác suất cao nhất có thể được lựa chọn.

1.5.6 Cú pháp

1.5.7 Từ nằm ngoài kho từ vựng

Hệ thống dịch máy thống kê lưu trữ các cụm từ một cách độc lập, không có mối quan hệ nào giữa các cụm từ. Những cụm từ không có trong dữ liệu sẽ không được dịch. Vấn đề này sẽ gặp phải khi thiếu dữ liệu, hoặc hệ thống được sử dụng trong lĩnh vực kiến thức mới.

CHƯƠNG II: DỊCH MÁY THỐNG KÊ TRÊN CƠ SỞ CỤM TỪ

Như đã nói ở trên cơ sở của việc dịch máy thống kê dựa bao gồm :

- ✓ Dịch máy thống kê dựa trên cơ sở từ
- ✓ Dịch máy thống kê dựa trên cơ sở cụm từ
- ✓ Dịch máy thống kê dựa trên cơ sở cú pháp

Ở phần này ta chỉ xét việc dịch máy trên cơ sở cụm từ.

2.1-Định nghĩa

Cụm từ (phrase) là một nhóm từ kết hợp với nhau tạo thành nghĩa nhưng không đầy đủ.

Cụm từ được phân thành mấy loại sau đây:

- ➢ Cụm danh từ
- ➢ Cụm động từ
- ➢ Cụm tính từ
- ➢ Cụm trạng từ

…….

Cụm từ có thể đóng nhiều vai trò trong câu như:

- ➢ Chủ ngữ : That girl is my cousin
  Cô gái ấy là chị tôi
- ➢ Tân ngữ: I bought a new English-VietNam dictionary yesterday
  Hôm qua tôi đã mua một cuốn từ điển Anh-Việt mới
- ➢ Bổ ngữ : She became a good student after taking that course
  Cô bé đã trở thành học sinh giỏi sau khi tham gia khóa học đó.
- ➢ Từ bổ nghĩa : The pub we often go to is awayls crowded
  Quán rượu chúng tôi thường đến luôn đông .

2.2-Mục đích của việc dịch máy thống kê trên cơ sở cụm từ

Mục đích chính của việc sử dụng cụm từ trong dịch máy thống kê là để giảm bớt hạn chế của việc dịch máy thống kê trên cơ sở từ..

Thông thường với một ngôn ngữ nhất định 1 từ có thể có nhiều nghĩa trong những văn cảnh khác nhau..Việc dịch máy dựa vào dịch từng từ một và sau đó

ghép tổ hợp của chúng với nhau thường dẫn đến những kết quả không tốt và phải xử lý một tổ hợp kết quả khá lớn

Ví dụ : Xét một câu đơn có n từ : $A_nA_{n-1}….A_2A_1$

Với mỗi từ $A_n,A_{n-1}…A_1$ sẽ có tương ứng $X_n,X_{n-1},X_{n-2}…X_1$ nghĩa

Do vậy với việc dịch trên cơ sở từ thì số ngôn ngữ đích tối đa có thể có sẽ là :

(Số Ngôn Ngữ) =$\sum_{i=1}^{n} Xi$ (chưa sử dụng các thuật toán tối ưu và nén với từ)

Việc sử dụng cụm từ trong dịch máy sẽ làm tăng độ chính xác của dịch máy đồng thời làm giảm đáng kể thời gian dịch của máy.

2.3 -Bảng cụm từ trong dịch máy thống kê

Dịch máy thống kê dựa trên cơ sở của bộ ngữ liệu có sẵn và từ đó tính xác suất để xuất hiện các từ liên quan .

Đối với dịch máy thống kê trên cơ sở cụm từ cũng vậy,ta cũng cần phải có một bộ ngữ liệu liên quan đến các cụm từ .Chính vì vậy bảng cụm từ (phrase tables) đã được xây dựng .

Bảng cụm từ(phrase-tables) được sử dụng trong dịch máy thống kê dựa trên cụm từ là rất lớn. Kích thước của chúng là một hệ quả trực tiếp của cách tiếp cận bảng cụm từ trong dịch máy thống kê (PB-SMT) sao cho sự tiên đoán trước có thể truy cập được một cách hiệu quả.

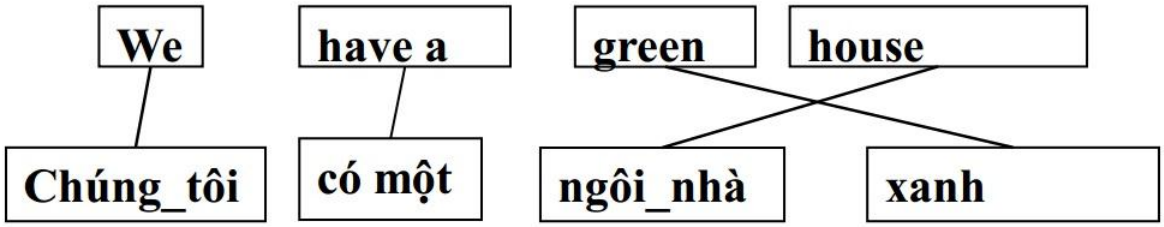
Việc tính toán trước sẽ làm gia tăng lên tổ hợp của cụm từ và cụm từ dư thừa cho bất kỳ cụm từ nào và tất cả các cụm từ con (Subphrase) có thể có được trong bảng cụm từ.

Bảng cụm từ được lưu trữ một cách rõ ràng hiện nay là đại diện được sử dụng rộng rãi nhiều nhất các mô hình dịch trong PB-SMT.

Phương pháp được sử dụng trong việc thực hiện tối ưu bảng cụm từ (Junczys-Dowmunt,2012a,b) cho Moses(Koehn 2007) có thể được sử dụng để thay thế cho các bảng cụm từ nhị phân hiện tại .

2.4-Quy trình dịch máy thống kê trên cơ sở cụm từ.

Đơn vị dịch : Cụm từ, là một chuỗi các từ liên tiếp bất kỳ



- Mỗi cụm tiếng Việt v_j ứng với một cụm tiếng Anh e_i

  $\phi(e_i|v_j)$: xác suất dịch cụm từ

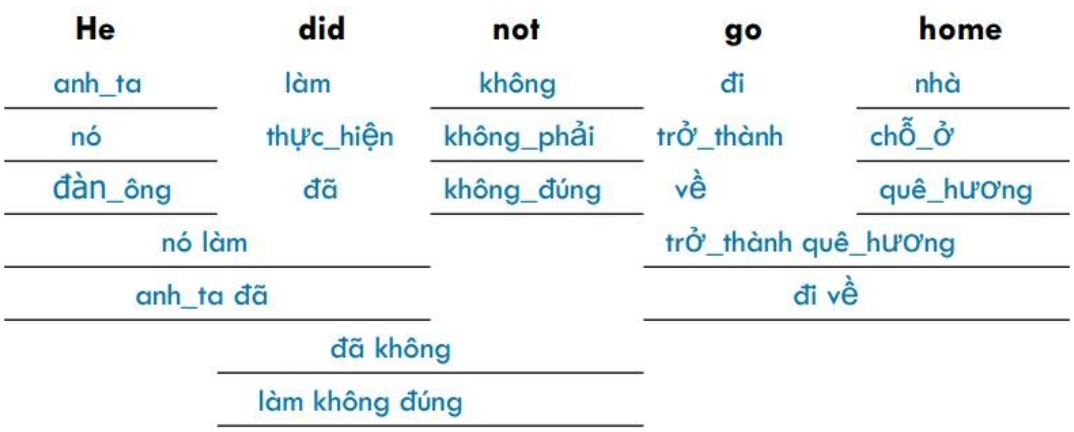
- Các cụm từ có thể bị dịch chuyển :
  - + $d(start_i\text{-}end_{i-1}\text{-}1)$: xác suất chuyển dịch
  - + $start_i$: vị trí đầu tiên của cụm từ tiếng Anh ứng với v_i
  - + end_{i-1}: vị trí cuối của cụm từ tiếng Anh ứng với v_{i-1}

->Xác suất p(e|v) :

$$p(e|v)=\prod_{i=1}^{i} \phi(ei|vj)d(start(i)-end(i-1)-1)$$

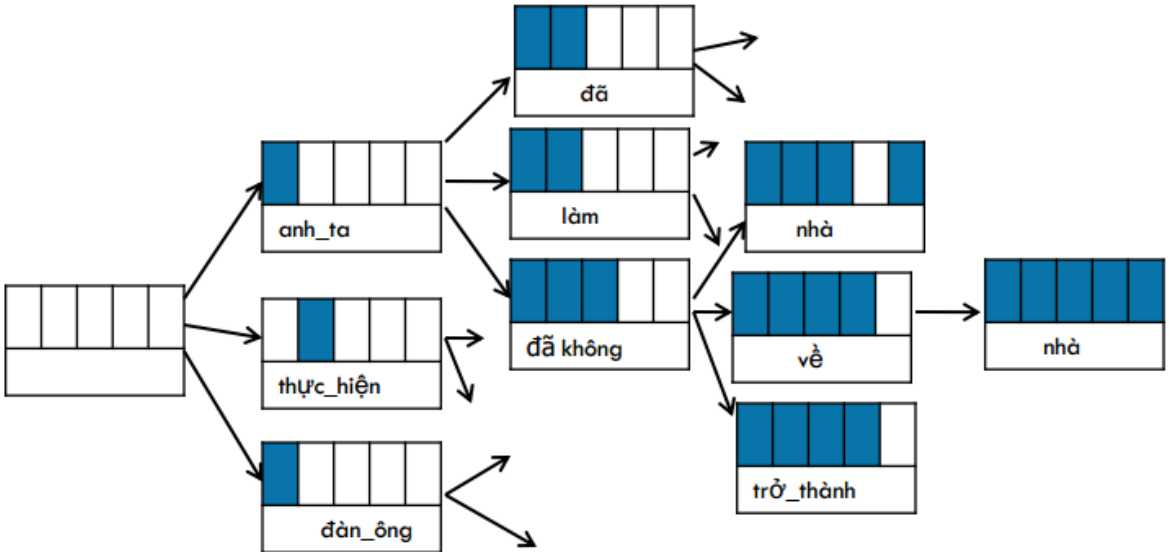
* Giải mã

Từ các khả năng dịch ta tìm ra được câu dịch tốt nhất.

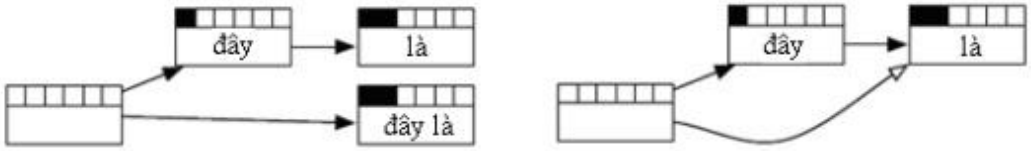


Mở rộng không gian giả thuyết

He	Did	Not	Go	Home



Giảm bớt số lượng giả thuyết.



CHƯƠNG III: PHƯƠNG PHÁP TỐI ƯU HÓA BẢNG CỤM TỪ DỰA VÀO CỤM

Bảng cụm từ được sử dụng trong dịch máy thống kê dựa trên cụm từ (PB-SMT) có kích thước rất lớn.Bảng cụm từ được lưu trữ một cách rõ ràng để sử dụng trong mô hình dịch.Mục đích của phương pháp sẽ trình bày dưới đây là để mô tả mệnh đề ,phương pháp mã hóa làm giảm dữ liệu từ, từ đó giảm đáng kể thời gian dịch máy nhưng vẫn đảm bảo chất lượng dịch một mức nhất định .Phương pháp mã hóa cụm (PR-Enc) nhằm mục đích giảm sự dư thừa bằng cách khai thác bảng cụm từ như một từ điển nén,sử dụng các mối quan hệ tịnh tiến .

Zens và Ney(2007) đã diễn tả thuộc tính của bảng cụm từ ở dạng nhị phân với bảng cụm từ của Mose là cơ sở.

Hầu hết tập dữ liệu của các ngôn ngữ chiếm bộ lớn khá lớn trong khi đó không gian đĩa cứng có hạn ,yêu cầu về tải trên đường truyền nên việc tối ưu hóa,rút gọn được nhiều người quan tâm và phát triển

Germann et al(2009) mở đầu cho việc đóng gói chặt chẽ (TBT) ,họ sử dụng các mô hình ngôn ngữ ,bảng cụm từ trong hệ thống SMT Portage(Sadat et al.,2005).TBT được lưu trữ trong mảng byte với mã hóa byte để làm giảm bớt không gian yêu cầu.

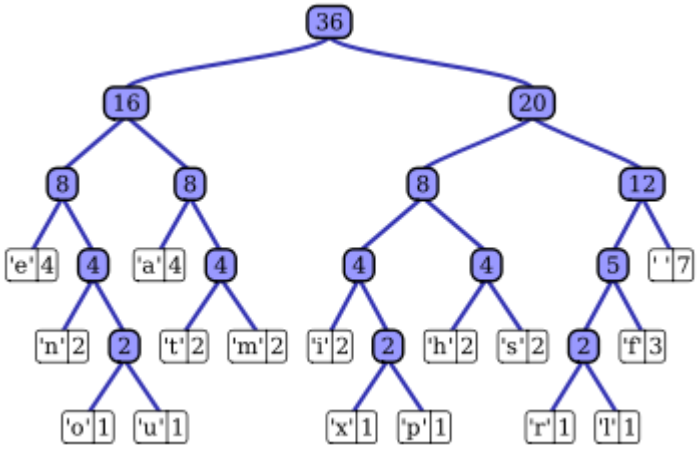
3.1 Nén ngữ liệu song ngữ

Conley và Klein(2008) đã đề xuất một chương trình mã hóa dữ liệu ngôn ngữ mục tiêu dựa trên cở sở liên kết từ và các mối quan hệ tịnh tiến.

Cụm từ mục tiêu sẽ được thay thế point-ers với các chỉ số bắt đầu và kết thúc của cụm từ tương ứng,chỉ số bản dịch và một con trỏ số nguyên cho mỗi mục tiêu.

Nén được thực hiện bằng việc sử dụng mã hóa Huffman.

Trong khoa học máy tính và lý thuyết thông tin, mã hóa Huffman là một thuật toán mã hóa dùng để nén dữ liệu. Nó dựa trên bảng tần suất xuất hiện các kí tự cần mã hóa để xây dựng một bộ mã nhị phân cho các kí tự đó sao cho dung lượng (số bít) sau khi mã hóa là nhỏ nhất.



Cây Huffman sinh ra từ xâu kí tự "this is an example of a huffman tree". Tổng số bit để mã hóa là 135, không kể các kí tự trống.

Gần đây nhất Sanchez-Martinez et al. (2012) đã đề xuất sử dụng "bi-words" để nén dữ liệu song ngữ,tương tự như của Conley và Klein(2008) các quan hệ tịnh tiến và phương pháp mã hóa Huffman được sử dụng để cải thiện dữ liệu được mã hóa.

Tóm tắt về giải thuật mã hóa HuffMan-Cây Huffman

Giải thuật :

Trong giải thuật giải bài toán xây dựng cây mã tiền tố tối ưu của Huffman, ở mỗi bước ta chọn hai chữ cái có tần số thấp nhất để mã hóa bằng từ mã dài nhất. Giả sử có tập A gồm n ký hiệu và hàm trọng số tương ứng **W**(i),i= 1…n

- Khởi tạo: Tạo một rừng gồm n cây, mỗi cây chỉ có một nút gốc, mỗi nút gốc tương ứng với một kí tự và có trọng số là tần số/tần suất của kí tự đó **W**(i),.
- Lăp:
  - Mỗi bước sau thực hiện cho đến khi rừng chỉ còn một cây:
  - Chọn hai cây có trọng số ở gốc nhỏ nhất hợp thành một cây bằng cách thêm một gốc mới nối với hai gốc đã chọn. Trọng số của gốc mới bằng tổng trọng số của hai gốc tạo thành nó.

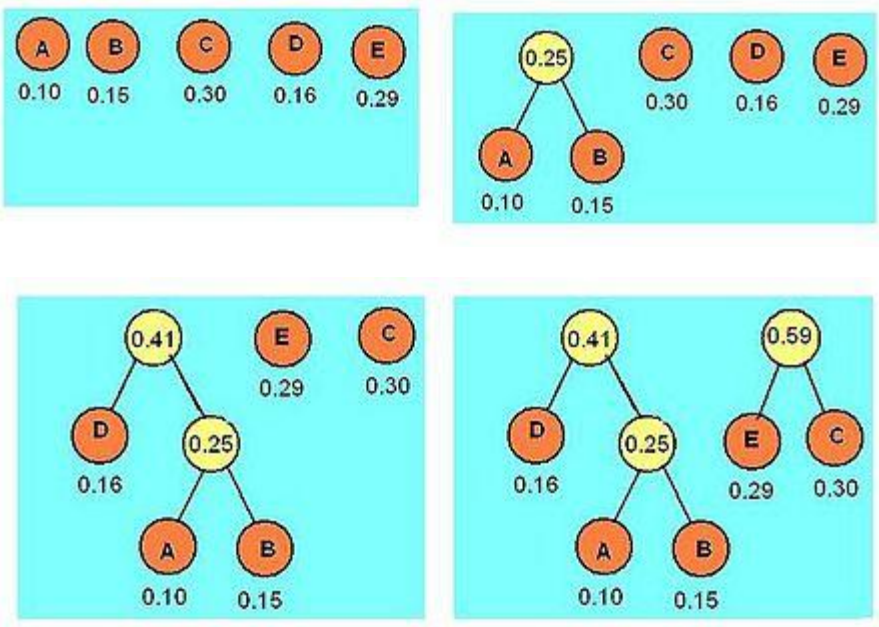
Như vậy ở mỗi bước số cây bớt đi một. Khi rừng chỉ còn một cây thì cây đó biểu diễn mã tiền tố tối ưu với các ký tự đặt ở các lá tương ứng.

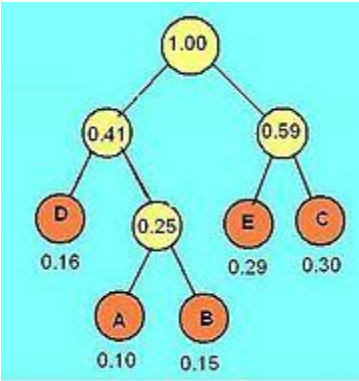
Ví dụ :

Với bảng tần suất của 5 chữ cái A,B,C,D,E như sau tương ứng là 0.10; 0.15; 0.30; 0.16; 0.29

A	B	C	D	E
0.10	0.15	0.30	0.16	0.29

Quá trình xây dựng cây Huffman diễn ra như sau :





Như vậy bộ mã tối ưu tương ứng là :

A	B	C	D	E
010	011	11	00	10

3.2 Nén bảng cụm từ

Junczys-Dowmunt (2012a) giới thiệu một kiến trúc làm nhỏ bảng cụm từ với việc sử dụng PR-Enc. Cơ bản của việc nén này là sử dụng thuật toán Simple-9 (Anh and Moffat, 2004)(S9),mã hóa biến-byte (Scholer et al., 2002) và mã hóa Huffman (Huffman, 1952) của các từ mục tiêu,điểm và các điểm liên kết.

Để giảm được kích thước cho các cụm từ nguồn ta sử dụng một hàm băm với các chỉ số.

Juncczys-Dowmunt (2012b) mô tả việc tối ưu của chỉ số cụm từ và tác động của nó đến chất lượng bản dịch.

Việc thực hiện tối ưu đã đạt được một kết quả khá tốt với hơn 77% khi so sánh với bảng cụm từ nhị phân Moses với hiệu suất tốt hơn đáng kể.

Mô tả nén sử dụng thuật toán Simple-9:

Simple9(S9) là một thuật toán đạt được kết quả nén tốt hơn nhiều mã hóa biến byte và cũng đã cải thiện được tốc độ nén.S9 không phải là liên kết byte nó là sự liên kết giữa các từ và bit liên kết.

Ý tưởng cơ bản của S9 là cố gắng để đóng gói giống như số nguyên vào một trong 32-bit từ.

Để làm được điều này ,S9 chia mỗi từ thành 4 bit trạng thái và 24 bít dữ liệu.Ví dụ như nếu 7 giá trị tiếp theo mà tất cả chúng đều nhỏ hơn 16,thì chúng ta có thể lưu trữ chúng như dạng 7 giá trị 4-bit,hoặc nếu sau 3 giá trị tiếp theo mà chúng nhỏ hơn 512 thì chúng ta có thể lưu trữ chúng dưới dạng 3 giá trị 9-bit (để lại một bit không dùng)

Ví dụ về việc nén string:

Giả sử ta có 1 đoạn string với 8 ký tự (example : “abcdefgh”)..Nếu chúng ta đặt vào kiểu mảng thì chúng ta sẽ có mảng với 8 byte .

| a=1 | b=2 | c=3 | d=4 | e=5 | f=6 | g=7 | h=8 |

ở dạng ASCII

|97|98|99|100|101|102|103|104|

Dạng bit nhị phân tương ứng sẽ là :

00000001|00000010|00000011|00000100|00000101|00000110|00000111|00001000

Với mỗi byte(8bit) thì chúng ta có thể lưu trữ được 2^8=256 ký tự,tuy nhiên trên thực tế với ngôn ngữ giao tiếp thì số ký tự thường chỉ ở mức 24-28 chữ cái(đối với ngôn ngữ la-tinh) do vậy thay vì sử dụng 8bit ta chỉ cần tới 5bit(tối đa 2^5=32) là có thể lưu trữ được 1 chữ cái.

Chuyển về dạng 5 bit thì ta sẽ cắt bỏ 3 bit đầu đi và kết quả sẽ là

00001|00010|00011|00100|00101|00110|00111|01000

Ta có thể gộp dạng trên về dạng :

00001000|10000110|01000010|10011000|11101000

Như vậy ta đã chuyển từ 8 byte về dạng 5 byte trong việc mã hóa string : abcdef

3.3 Mã hóa thứ hạng cụm

Ý tưởng chung của việc mã hóa cụm tương tự như các phương pháp nén dựa trên từ điển cổ điển..Các cụm từ con (subphrases) lặp đi lặp lại được thay thế bằng con trỏ đến các S9 trong từ điển.

Việc giải nén dựa trên việc tìm kiếm và chèn lại của các subpharses trong ký tự con trỏ.

Nếu chúng ta đơn giản hóa những phương thức trên bằng việc xóa tất cả các dữ liệu yêu cầu bên ngoài và chuyển nó tới bảng pharse,chúng ta sẽ có dạng cơ bản của PR-Enc.Thay vì việc nén một bitext với một từ điển các cụm từ,chúng ta nén từ vựng riêng của mình.

Việc mã hóa thứ hạng cụm sẽ chia sẻ thuộc tính với mã hóa thứ hạng từ và chỉ ra phương thức nén bitext.

Phương pháp nén được thực hiện với một cách khác : Dữ liệu tuần tự được biểu diễn dạng đồ thị cấu trúc giống như dạng cây hoặc là máy tự động.

Mã hóa thứ hạng cụm cũng có thể sử dụng để biểu diễn dữ liệu mục tiêu từ bảng cụm từ nhị phân Moses dựa trên Zens và Ney(2007) vào một cấu trúc đồ thị .

3.4 Thủ tục mã hóa (Encoding Procedure)

Dưới đây là giải thuật mã hóa thủ tục.

Function EncodeTargetPhrase($\mathbf{s}, \mathbf{t}, A, \text{Order}, \text{RankedPT}$)
  $\hat{\mathbf{t}} \leftarrow \langle\,\rangle$; $\hat{A} \leftarrow A$
  $P \leftarrow \{\langle i, j, m, n\rangle : (0 \le i < i + m \le |\mathbf{s}| \wedge 0 \le j < j + n \le |\mathbf{t}|)$
    $\wedge\, \forall \langle i', j'\rangle \in A : (i \le i' < i + m \Leftrightarrow j \le j' < j + n)$
    $\wedge\, (m < |\mathbf{s}| \vee n < |\mathbf{t}|)\}$
  $Q \leftarrow \text{Queue}(P, \text{Order})$
  while $|Q| > 0$ **do**
    $\langle i, j, m, n\rangle \leftarrow \text{Pop}(Q)$
    $\mathbf{s}' \leftarrow \text{Substring}(\mathbf{s}, i, m)$
    $\mathbf{t}' \leftarrow \text{Substring}(\mathbf{t}, j, n)$
    if $\exists r : \langle \mathbf{s}', \mathbf{t}', r\rangle \in \text{RankedPT}$ **then**
      $T[j] \leftarrow \langle i - j, |\mathbf{s}| - (i + m), r\rangle$
      $S[j] \leftarrow n$
      $\hat{A} \leftarrow \hat{A} \setminus \{\langle i', j'\rangle \in \hat{A} : i \le i' < i + m \wedge j \le j' < j + n\}$
      $P \leftarrow P \setminus \{\langle i', j', m', n'\rangle : i \le i' < i + m \vee j \le j' < j + n$
        $\vee\, i' \le i < i' + m' \vee j' \le j < j' + n'\}$
      $Q \leftarrow \text{Queue}(P, \text{Order})$
  $j \leftarrow 0$
  while $j < |\mathbf{t}|$ **do**
    if $S[j] > 0$ **then**
      $\hat{\mathbf{t}} \leftarrow \hat{\mathbf{t}} \cdot \langle T[j]\rangle$
      $j \leftarrow j + S[j]$
    else
      $\hat{\mathbf{t}} \leftarrow \hat{\mathbf{t}} \cdot \langle t_j\rangle$
      $j \leftarrow j + 1$
  return $\langle \hat{\mathbf{t}}, \hat{A}\rangle$

Figure 1. Algorithm for Phrasal Rank-Encoding

Mã hóa yêu cầu từ bảng cụm từ để có những thông tin liên kết từ.Để thực hiện việc mã hóa có hiệu quả nó có thể tìm các cặp cụm từ liên kết và lấy lại thứ hạng của cụm từ mục tiêu liên quan tới cụm từ nguồn tương ứng.

Danh sách được sắp xếp giảm dần của xác suất dịch P,tức là bản dịch tốt nhất sẽ có cấp bậc là 0,bảng dịch chất lượng càng kém thì có cấp bậc càng cao.

Bảng cụm từ tìm kiếm được thông qua các thuật toán như RankedPT.

Chúng ta minh họa thuật toán RankedPT nêu trên với ví dụ sau:

Cho một câu Spanish –English với các cặp cụm từ đã được dóng.(Việc dóng được mô tả trong các hộp hình 2)

es: Maria no daba una bofetada a la bruja verde

en: Mary did not slap the green witch

Các cặp cụm từ này được đại diện bởi các quadruple(1 bộ 4 chỉ số) bao gồm các chỉ số : vị trí cụm từ nguồn, vị trí cụm từ đích, chiều dài cụm từ nguồn, chiều dài cụm từ đích.

Trong dòng 3 của thuật toán các cặp câu được dóng đúng sẽ tính toán,kết quả cho trong hình sau bởi những hình chữ nhật trống hoặc được phủ đen.

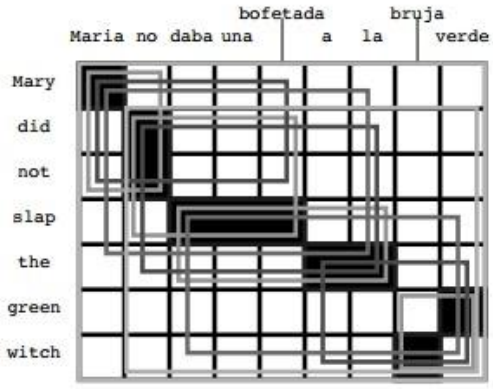


Figure 2. Archetypical example for phrase pair extraction by Knight and Koehn (2003)

các cặp cụm từ hoàn chỉnh bị loại bỏ vì nó không là các cặp cụm từ con được dóng đúng điều kiện đầu tiên của biểu thức, ở dòng 3 yêu cầu các cặp cụm từ con nằm trong đường bao của các cụm mã hóa.

Zens và các cộng sự (2002) định nghĩa các cụm từ con phù hợp với việc dóng cơ bản với thủ tục tương tự được sử dụng trong suốt quá trình rút các cặp cụm từ khi mô hình dịch được tạo để tránh việc tự tham chiếu.

Các cụm từ con được chèn vào hàng đợi (dòng 4) dựa theo thứ tự :”các cụm con được đánh thứ tự giảm dần theo chiều dài,vị trí bắt đầu dịch và sau đó đến chiều dài và vị trí bắt đầu của cụm nguồn.”

Với ví dụ trên cặp cụm từ nguồn là hàng đợi :

es: no daba una bofetada a la bruja verde

en: did not slap the green witch

Nó sẽ được kiểm tra để đưa vào bảng xếp hạng cụm từ(dòng 9) với thứ hạng tính từ 0.

Cụm từ đích được thay thế bởi 1 ký hiệu con trỏ

es: Maria no daba una bofetada a la bruja verde

en: Mary (0,0,0)

Các giá trị nguyên của bộ 3 này được hiểu như sau :

- Đầu tiên là sự khác nhau giữa vị trí nguồn và đích của cặp cụm con.

- Thứ 2 là khoảng của đường bao cụm nguồn con bên phải từ cuối cụm mã hóa.

- Giá trị cuối cùng là thứ hạng của cặp cụm con được chọn.

Tất cả các điểm được dóng nằm trong đường bao của cặp cụm con được chọn được loại bỏ(dòng 12) và tất cả những cặp cụm con chồng lên cặp cụm con đang xét được xóa khỏi hàng đợi (dòng 13,14)

Chỉ 1 cặp cụm từ còn lại trong hàng đợi là :

Es: Maria

En: Mary

Áp dụng thủ tục tương tự như trên

Các cụm mã hóa sau được tạo ra :

es: Maria no daba una bofetada a la bruja verde

en: (0,8,0) (0,0,0)

các cụm con đích mà không được thay thế được tìm thấy sẽ được giữ lại như những từ plain.

3.5 Thủ tục giải mã (Decoding Procedure)

Một thủ tục giải mã đơn giản xử lý cây chủ yếu là cây nhị phân với thời gian mũ.Tuy nhiên nếu xem xét tất cả các cụm đích ứng với một câu,một thuật toán quy hoạch động với độ phức tạp tuyến tính cho mỗi cụm có thể được xây dựng.

Moses truy vấn bảng cụm từ xử lý các câu theo kiểu từ trái sang phải bắt đầu với các cụm từ con có chiều dài 1 và tăng dần chiều dài của nó khi đạt đến giới hạn ,sau đó chuyển sang từ tiếp theo với chiều dài lại là 1.

Do đó ,nếu 1 cụm từ được tìm thấy thì các tiền tố của nó đã được xử lý trước đó.

Nếu tất cả các cụm từ truy vấn được cache lại cho việc giải mã và tất cả các cụm được sử dụng cho việc giải mã được cache lại cho việc tìm kiếm thì tổng số bảng cụm từ truy cập là tương tự như trong một bảng cụm từ tuyến tính.

Với việc caching một cụm từ đích cho cụm nguồn :”Maria no daba una bofetada” sẽ được tìm thấy ngay lập tức mà không cần duyệt các nhánh còn lại.

Các cụm từ con "a la" sẽ vẫn được xử lý,nhưng khi Môi-se truy vấn cụm từ đó, nó sẽ được lấy từ bộ nhớ cache.

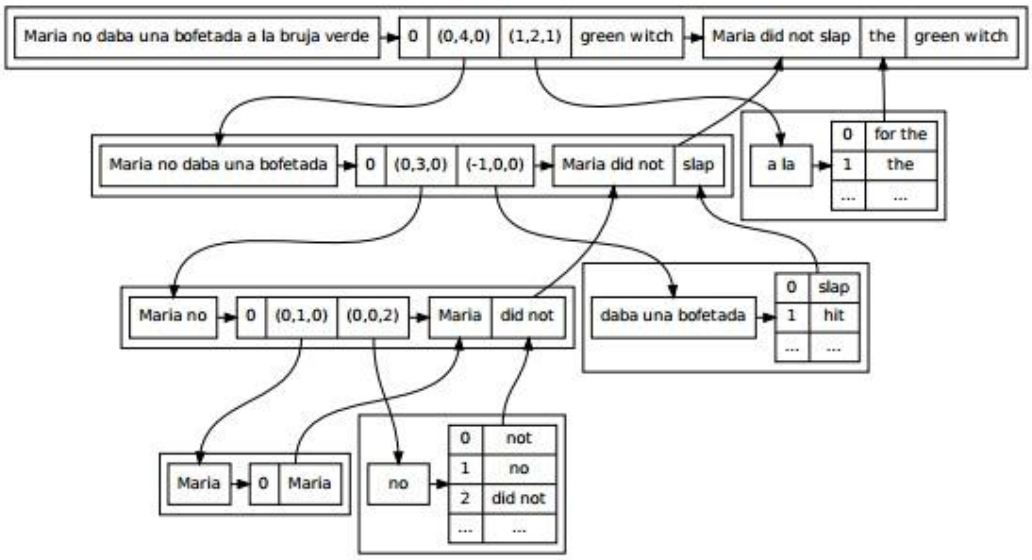


Figure 3. Phrasal Rank-Decoding without Caching

Thuật toán giải mã :


```
1  Function GetTargetPhraseCollection(s)
2      T ← ⟨ ⟩; r ← 0
3      while r < NumberOfTargetPhrases(s) do
4          if InCache(s, r) then
5              ⟨t, A⟩ ← GetFromCache(s, r)
6          else
7              ⟨t̂, Â⟩ ← GetFromPhraseTable(s, r)
8              ⟨t, A⟩ ← DecodeTargetPhrase(s, r, t̂, Â)
9          T ← T · ⟨⟨t, A⟩⟩
10         r ← r + 1
11     return T
12 Function DecodeTargetPhrase(s, r, t̂, Â)
13     t ← ⟨ ⟩; A ← Â
14     j ← 0
15     while j < |t̂| do
16         if Type(t̂_j) = Pointer then
17             ⟨k, l, r′⟩ ← t̂_j
18             i ← k + |t|
19             m ← |s| − l + 1
20             s′ ← Substring(s, i, m)
21             if InCache(s′, r′) then
22                 ⟨t′, A′⟩ ← GetFromCache(s′, r′)
23             else
24                 ⟨t̂′, Â′⟩ ← GetFromPhraseTable(s′, r′)
25                 ⟨t′, A′⟩ ← DecodeTargetPhrase(s′, r′, t̂′, Â′)
26             t ← t · t′
27             A ← A ∪ {⟨i + i′, j + j′⟩ : ⟨i′, j′⟩ ∈ A′}
28         else if Type(t̂_j) = Word then
29             t ← t · ⟨t̂_j⟩
30         j ← j + 1
31     AddToCache(s, r, ⟨t, A⟩)
32     return ⟨t, A⟩
```

Figure 4. Retrieving a set of target phrases

Mọi tập các cụm từ đích cho 1 câu nguồn được tạo .Nếu một cụm đích với thứ hạng cho trước được quan sát thấy trước đó nó sẽ được phục hồi từ cache.Với trường hợp khác nếu một phiên bản giải mã được chuyển từ bảng cụm từ tới DecodeTargetPharse.

Cụm con đích giải mã sau đó được nối với cụm đích hiện tại và điểm dóng cụm được thêm vào. Dóng đầu ra được dịch chuyển phù hợp ,kết quả được cache lại .

CHƯƠNG IV: THỰC NGHIỆM TỐI ƯU HÓA CỤM TỪ BẰNG HỆ DỊCH MÁY THỐNG KÊ MOSES

Moses là một hệ thống dịch máy thống kê cho phép chúng ta tự đào tạo mô hình dịch cho cặp câu song ngữ.Tất cả những điều chúng ta cần là thu thập các bản dịch song ngữ.

- ✓ Có khả năng tự động huấn luyện các mô hình dịch
- ✓ Input : bộ dữ liệu song ngữ
- ✓ Thuật toán tìm kiếm : Tìm ra bản dịch tốt nhất có thể

4.1 Cài đặt hệ thống Moses

Hệ thống dịch máy moses có thể cài đặt trên các os khác nhau như Linux,OSX hay Windows .Ở phần demo này chúng ta sẽ cài đặt và chạy các test case trên Linux cụ thể là Ubuntu phiên bản 11.10

Các công cụ đi theo :

- ✓ Hệ thống đã cài Boost
- ✓ SRILM

Công cụ xây dựng mô hình dịch : GIZA++,mkcls

4.2 Các bước để chạy bộ công cụ và sử dụng bản dữ liệu thực nghiệm.

- ➢ Chuẩn hóa dữ liệu
- ➢ Xây dựng mô hình ngôn ngữ
- ➢ Xây dựng mô hình dịch
- ➢ Dịch máy
- ➢ Đánh giá kết quả dịch

4.2.1 Chuẩn hóa dữ liệu

Dữ liệu đầu vào cần được chuẩn hóa theo đúng dạng qui định

Việc chuẩn hóa dữ liệu có thể bao gồm những công việc như :

- Tách từ
- Tác câu
- Chuyển sang chữ thường,chữ hoa
- Loại bỏ từ dư thừa
- ….

4.2.2 Xây dựng mô hình ngôn ngữ

SRILM là một gói công cụ để xây dựng mô hình dịch ngôn ngữ.

Nó giúp chúng ta xây dựng được mô hình ngôn ngữ trước khi cho vào máy dịch

4.2.3 Xây dựng mô hình dịch

Sử dụng GIZA++ để xây dựng mô hình dịch và dùng mkcls để ước lượng giá trị cực đại cho mỗi mô hình.

4.2.4 Dịch máy

Sau khi đã xây dựng mô hình ngôn ngữ,mô hình dịch và chuẩn hóa dữ liệu

Việc tiếp theo của chúng ta là dịch máy .

Bộ ngữ liệu song ngữ trên các cặp ngôn ngữ khác nhau .

Tiến hành dịch và so sánh kết quả.

4.2.4 Đánh giá kết quả dịch

Kết quả dịch máy thống kê có chính xác hay không đều dựa vào các chỉ số dịch máy .Có 2 chỉ số cần quan tâm đó là chỉ số BLEU và chỉ số NIST.

a.Chỉ số BLEU

Đây là chỉ số đánh giá chất lượng dịch của máy dịch thống kê từ ngôn ngữ này sang ngôn ngữ khác.

Kết quả dịch máy thống kê càng chính xác thì chỉ số BLEU càng cao và ngược lại .Điểm chỉ số BLEU được tính dựa vào việc so sánh câu dịch được với một tập hợp các câu dịch tốt,sau đó lấy giá trị trung bình từ những câu này .

Chỉ số BLEU có giá trị nằm từ 0 đến 1.Chỉ số càng gần 1 thì chất lượng dịch càng tốt,chỉ số càng nhỏ gần tới 0 thì chất lượng dịch càng kém .

b. Chỉ số NIST

Về cơ bản phương pháp đánh giá nhờ chỉ số NIST cũng tương tự như chỉ số BLEU nhưng nó cũng có một số khác biệt

Chỉ số NIST cung cấp thông tin cần thiết để đánh giá trọng số dịch.

4.3 DEMO và báo cáo kết quả thực nghiệm

Cấu hình phần cứng và phần mềm cài đặt .

-CPU Dual Core 2.27Ghz

-RAM 2GB

-Hệ điều hành Ubuntu 11.10

Các công cụ hỗ trợ :

-Giza++

-Srilm

-Boost

Tài liệu tham khảo

1- Phrasal Rank-Encoding: Exploiting Phrase Redundancy and Translational Relations for Phrase Table Compression (Marcin Junczys-Dowmunt)

2- Statistical Machine Translation: Hướng dẫn cài đặt và sử dụng MOSES,

http://www.statmt.org/moses_steps.html

3- Sử dụng một số bài viết trên trang http://www.wikipedia.org

4-Ngoài ra luận văn còn sử dụng tài liệu trên các trang mạng khác.