Tạp chí Tin học và Điều khiển học, T.27, S.3 (2011), 206–217

NGHIÊN CỨU TỪ VỰNG TIẾNG VIỆT VỚI HỆ THỐNG SKETCH ENGINE

PHAN THỊ HÀ[1], NGUYỄN THỊ MINH HUYỀN[2], LÊ HỒNG PHƯƠNG[2], ADAM KILGARRIFF[3], SIVA REDDY [4]

[1]*Học viện Công nghệ Bưu chính Viễn thông*
[2]*Đại học Quốc gia Hà Nội*
[3] *Lexicography MasterClass and ITRI, University of Brighton, UK*
[4] *IIIT Hyderabad, India*

Tóm tắt. Skech Engine là một hệ thống cho phép truy vấn kho ngữ liệu dựa vào tập quan hệ ngữ pháp của một ngôn ngữ nào đó, phục vụ cho việc nghiên cứu từ vựng học. Hệ thống này đã được sử dụng cho nghiên cứu từ vựng, đặc biệt là xây dựng từ điển của nhiều ngôn ngữ (Anh, Tiệp, Nhật, Trung, ...). Bài báo này sẽ giới thiệu hệ thống Sketch Engine và nghiên cứu triển khai hệ thống này cho tiếng Việt. Chúng tôi cũng trình bày cách thức xây dựng kho ngữ liệu và tập các quan hệ ngữ pháp cơ bản tiếng Việt để phục vụ cho hệ thống truy vấn kho ngữ liệu trong Sketch Engine.

Abstract. The Sketch Engine is a corpus query system based on grammatical relations of a language. This system has been widely used in lexicography, particularly for building dictionaries of different languages such as English, Japanese, Chinese, etc. This paper presents an approach to apply the Sketch Engine to Vietnamese. A method for building corpus and fundamental grammatical relations for Vietnamese is proposed for the corpus query system in Sketch Engine.

Từ khoá. Phác thảo từ, Sketch Engine, kho ngữ liệu, quan hệ ngữ pháp.

Keywords. Word sketch, Sketch Engine, corpus, grammatical relation.

1. GIỚI THIỆU

Sử dụng ngữ liệu văn bản để xây dựng từ điển là một phương pháp đã được áp dụng từ lâu. Khi chưa có máy tính, các nhà từ điển học sử dụng các tấm thẻ chỉ mục để lưu trữ thông tin sử dụng từ. Vào những năm 1980, cùng với dự án COBUILD (*Collins Birmingham University International Language Database*) nhằm xây dựng và phân tích kho văn bản tiếng Anh phục vụ việc xây dựng từ điển, Sinclair [13] đã nhìn thấy khả năng lưu trữ, sắp xếp, tìm kiếm một cách khách quan hơn của máy tính so với con người. Kể từ dự án này, các nhà xây dựng từ điển sử dụng công cụ truy vấn kho ngữ liệu, cho phép tra cứu từ khóa trong ngữ cảnh để nghiên cứu hành vi của một từ. Do vậy, các hệ thống truy vấn kho ngữ liệu (Corpus Query Systems - CQSs) đóng vai trò quan trọng trong lý thuyết và thực hành biên soạn kho từ điển. Các nhà nghiên cứu từ điển sử dụng hệ thống truy cập vào kho ngữ liệu để tìm kiếm các cụm từ, thứ tự ưu tiên của các từ xung quanh một từ, các mẫu ngữ pháp, để sắp xếp các từ đi cùng theo nhiều tiêu chí khác nhau, để xác định các kho ngữ liệu con cho việc tìm kiếm. Có thể kể đến một số hệ thống truy vấn kho ngữ liệu như WordSmith, MonoConc, Stuttgart workbench hay Manatee.

Tuy nhiên, khi kích thước kho văn bản ngày càng khổng lồ, số ngữ cảnh xuất hiện một từ trở nên quá lớn, thì công cụ tìm kiếm ngữ cảnh đơn giản trở nên không đủ. Church. K. W. và Hanks. P [21] đã khởi xướng lĩnh vực thống kê từ vựng. Họ đề xuất sử dụng thông tin tương hỗ (*mutual information*) để đo tính trội (*salience*) về quan hệ giữa hai từ. Nếu ta tìm tất cả các từ xuất hiện trong lân cận 5 từ của một từ nào đó trong kho văn bản, sau đó tính toán tính trội của mỗi từ này với từ mà ta quan tâm, thì ta có thể tổng hợp dữ liệu văn bản cho từ đó bằng một danh sách các từ cùng xuất hiện (*collocates*) được sắp theo thứ tự tính trội của chúng. Cách tiếp cận này đã thu hút được mối quan tâm của các nhà làm từ điển và chức năng xác định các từ đồng xuất hiện có trong tất cả các công cụ truy vấn vấn kho ngữ liệu.

Bài báo đề xuất việc sử dụng một hệ thống truy vấn kho ngữ liệu để khai thác thông tin từ vựng tiếng Việt. Hệ thống được lựa chọn là Sketch Engine do nhóm nghiên cứu của Kilgarriff [5] phát triển, đã được sử dụng cho nhiều ngôn ngữ. Trong Mục 2 sẽ giới thiệu sơ bộ về hệ thống Sketch Engine. Mục 3 trình bày việc thu thập và tiền xử lí kho ngữ liệu tiếng Việt để sử dụng trong hệ thống này. Mục 4 giới thiệu về việc xây dựng tập luật biểu diễn quan hệ ngữ pháp phục vụ cho tra cứu cách sử dụng từ.

2. SKETCH ENGINE

Các công cụ truy vấn kho ngữ liệu hỗ trợ thống kê từ vựng thường bị ảnh hưởng bởi các vấn đề sau [4]:

- Sự thiếu cân bằng giữa các từ thông thường trong danh sách từ đồng xuất hiện so với các từ hiếm (ít xuất hiện trong kho ngữ liệu).
- Các danh sách từ thường bao gồm các dạng từ, tức là các từ đã biến đổi (hợp giống, số, v.v.) thay vì là các từ chuẩn (lemma).
- Việc quyết định xét bao nhiêu từ nằm bên trái hoặc bên phải một từ là ngẫu nhiên.
- Trong danh sách thường có nhiễu (những từ không đáng quan tâm về mặt ngôn ngữ học).
- Trong cùng một danh sách có thể có nhiều loại từ với vai trò hoàn toàn khác nhau như chủ ngữ của một động từ, bổ ngữ của động từ đó, trạng từ, phụ động từ hay giới từ...

Các hệ thống truy vấn kho ngữ liệu phổ biến đều đã giải quyết được vấn đề thứ nhất và thứ hai. Vấn đề thứ nhất là một trong các thống kê tính trội, các hệ thống truy vấn hiện đại có thể sử dụng một tham số để điều chỉnh tỉ suất đồng xuất hiện của các từ [3]. Tham số này có thể được chọn sẵn trong hệ thống hoặc cho phép người dùng lựa chọn. Vấn đề thứ hai liên quan tới việc xác định từ nguyên thể của văn bản, sau đó áp dụng các danh sách từ nguyên thể này thay vì các dạng từ biến đổi khác. Word Sketch, tiền thân của hệ thống Sketch Engine, có khả năng giải quyết ba vấn đề còn lại. Thay vì chỉ đưa ra tất cả các ngữ cảnh văn bản xung quanh một từ trong tiếng Anh, Word Sketch cho phép người sử dụng xem xét ngữ cảnh theo quan hệ ngữ pháp và cung cấp thống kê về tần suất xuất hiện các từ theo mỗi quan hệ ngữ pháp.

Word Sketch đã được Kilgarriff [5] phát triển thành hệ thống Sketch Engine - hệ thống có thể nhận đầu vào là kho ngữ liệu của bất cứ ngôn ngữ nào cùng với bộ mẫu ngữ pháp tương ứng. Ngoài chức năng của Word Sketch, hệ thống còn cung cấp thêm các chức năng:

– Thesaurus: cho phép tra cứu các từ đồng và phản nghĩa.

– Sketch Difference: cho phép so sánh thông tin của hai từ tương tự nhau.

Hiện thời, Skech Engine đã trở thành một hệ thống truy vấn kho ngữ liệu đã được thử nghiệm trên nhiều ngôn ngữ khác nhau (Anh, Séc, Nhật, Trung, Nga, Xlôven...) và được đánh giá là có hiệu quả tốt trong việc xây dựng từ điển, việc nghiên cứu và thực hành ngôn ngữ.

Đối với tiếng Việt, các nhà làm từ điển hiện nay thường mới chỉ có công cụ để tra cứu ngữ cảnh của một từ trong kho ngữ liệu, chưa có các thống kê tự động để so sánh, chọn lọc các ngữ cảnh. Việc sử dụng một bộ công cụ như hệ thống Sketch Engine sẽ là rất hữu ích, giúp cải thiện quy mô và chất lượng từ điển. Trong các phần tiếp theo, chúng tôi sẽ giới thiệu việc xây dựng kho ngữ liệu lớn từ Internet và bộ quan hệ ngữ pháp cho tiếng Việt tương thích với hệ thống Sketch Engine để có thể sử dụng hệ thống này cho nghiên cứu từ vựng tiếng Việt.

3. XÂY DỰNG VÀ TIỀN XỬ LÍ KHO NGỮ LIỆU TIẾNG VIỆT

Nghiên cứu từ vựng đòi hỏi xây dựng một kho ngữ liệu có kích thước càng lớn càng tốt. Trước kia, công việc này đòi hỏi khá nhiều thời gian và công sức. Chẳng hạn, để có một kho ngữ liệu tiếng Việt chứa khoảng 80 triệu âm tiết năm 2011, Trung tâm từ điển Vietlex đã bắt đầu công việc thu thập dữ liệu từ năm 1998[1].

Ngày nay, với sự bùng nổ của Internet, công việc xây dựng kho ngữ liệu đã trở nên dễ dàng và thuận lợi hơn nhờ việc tải các văn bản sẵn có từ các trang web. Cách làm này lần đầu tiên đã được thực hiện vào cuối những năm 1990 [17]. Grefenstette và Nioch [8] đã chỉ ra lượng dữ liệu rất lớn có trên Internet, kể cả với các ngôn ngữ ít phổ biến hơn. Baroni và Bernardini [12] cũng giới thiệu một công cụ mã nguồn mở cho việc thu thập dữ liệu từ Internet là công cụ BootCaT. Keller và Lapata [7] đã chứng tỏ tính hợp lệ của việc sử dụng các kho ngữ liệu Web cho nghiên cứu ngôn ngữ học bằng cách so sánh tự động cũng như thủ công các mô hình ngôn ngữ thu được từ kho ngữ liệu Web với các mô hình thu được từ kho ngữ liệu truyền thống. Việc thu thập dữ liệu Web lại có ưu điểm là cho phép cập nhật ngữ liệu thường xuyên, phát hiện những hiện tượng ngôn ngữ đa dạng và phong phú một cách khách quan hơn so với thu thập dữ liệu truyền thống.

Trong phần này, ta sẽ xây dựng một kho ngữ liệu tiếng Việt từ Web có kích thước lớn khoảng 100 triệu từ, gồm các văn bản thuộc tất cả các lĩnh vực trong cuộc sống, tiến hành tách từ và gán nhãn từ loại để có thể đưa vào sử dụng trong hệ thống Sketch Engine.

Công việc chuẩn bị dữ liệu để đưa vào hệ thống truy vấn ngữ liệu cho một ngôn ngữ được chia thành các bước chính như sau:

– Bước 1. Lựa chọn một danh sách các từ hạt giống có tần suất xuất hiện trung bình.

– Bước 2. Thu thập dữ liệu từ Web bằng cách sử dụng các từ hạt giống để tạo truy vấn thông qua các cổng tìm kiếm Yahoo và Google và tải các trang kết quả về.

– Bước 3. Làm sạch văn bản, loại bỏ các thông tin quảng cáo và các thông tin nhiễu khác.

– Bước 4. Loại bỏ các văn bản trùng lặp.

[1] http://www.vietlex.com

– Bước 5. Tách từ, chuẩn hóa và gán nhãn từ loại.

Phương pháp thu thập ngữ liệu sử dụng ở đây về cơ bản giống như phương pháp đã sử dụng cho tiếng Anh và một số ngôn ngữ phổ biến khác trong [18], [11] và [1]. Vấn đề quan trọng cần làm là lập danh sách từ hạt giống cho từng ngôn ngữ. Phương pháp luận chung cho việc lựa chọn từ hạt giống và kết quả cụ thể đối với các ngôn ngữ đã được nhóm nghiên cứu giới thiệu trong [6].

Sau đây là chi tiết các bước thu thập dữ liệu cho tiếng Việt.

3.1. Lựa chọn danh sách từ hạt giống

Các từ hạt giống đóng vai trò từ khóa tìm kiếm để thu về các văn bản của một ngôn ngữ. Đây phải là các từ đặc trưng cho ngôn ngữ, tức là có tần suất xuất hiện đáng kể, và có tính phân biệt so với các từ trong ngôn ngữ khác. Để có được tập từ hạt giống cho một ngôn ngữ bất kì, ta sử dụng nguồn ngữ liệu Wikipedia (Wiki) của ngôn ngữ đó để xác định và lựa chọn các từ hạt giống dựa trên tần suất xuất hiện của chúng trong kho ngữ liệu này.

3.1.1. Trích rút kho ngữ liệu từ Wiki

Để trích rút văn bản từ kho Wiki, ta thực hiện các bước sau:

– Tải về khối dữ liệu nén XML Wiki.

– Trích rút các trang XML (có chứa các thẻ Wiki) từ khối dữ liệu nén XML Wiki.

– Phân tích cú pháp các trang XML để loại bỏ các nhãn Wiki, thu được các trang XML thô.

– Trích rút văn bản thô từ các trang XML thô bằng cách sử dụng công cụ Wikipedia2text[2] (có chỉnh sửa đôi chút).

Đối với tiếng Việt, với 426 MB dữ liệu nén tải về từ Wiki, có thể thu được 750 MB văn bản thô.

Ta thấy rằng phần lớn các bài Wiki không chứa văn bản liên quan mà là các định nghĩa ngắn gọn, các tập hợp liên kết [3]. Những bài như thế thường có kích thước nhỏ và sẽ bị loại bỏ. Ide [14] và các cộng sự đã đưa ra một ước lượng số từ tối thiểu để nhận biết một mục bài có văn bản liên quan là 2000 từ. Do vậy, ta coi các tệp tin Wiki nếu có chứa văn bản liên quan thì phải có dung lượng lớn hơn 10 KB (mặc dù trong thực tế có những tệp lớn hơn 10KB cũng không chứa văn bản liên quan, tuy nhiên ảnh hưởng của chúng về mặt thống kê không lớn). Sau khi loại bỏ các tệp nhỏ hơn 10 KB, ta thu được 57 MB văn bản tiếng Việt (6.8 triệu âm tiết). Kho văn bản này được dùng để xây dựng danh sách tần suất các từ.

[2]http://evanjones.ca/software/wikipedia2text.html

[3]Các mục liên kết tới các mục khác hoặc các trang khác

3.1.2. Lập danh sách tần suất các từ

Để thu được danh sách tần suất từ kho ngữ liệu Wiki, chúng tôi thực hiện tách từ các văn bản trong kho ngữ liệu tiếng Việt. Chúng tôi sử dụng một danh sách từ tiếng Việt để nhận dạng từ và tính tần suất. Thuật toán đơn giản chúng tôi sử dụng là duyệt theo từng câu từ trái sang phải, chọn ranh giới từ sao cho từ thu được có nhiều âm tiết nhất có thể. Cách lựa chọn này rõ ràng không phải bao giờ cũng chính xác, nhưng sai số là chấp nhận được cho mục đích lập danh sách tần suất từ.

3.1.3. Lựa chọn từ hạt giống từ danh sách tần suất

Tiêu chí chọn từ hạt giống của mỗi ngôn ngữ là khác nhau, ví dụ với tiếng Hà Lan thì chỉ các từ có độ dài ít nhất là 5 kí tự là được lựa chọn. Đối với tiếng Việt thì độ dài của từ không phải là tiêu chí để lựa chọn, qua khảo sát các văn bản tiếng Việt cho thấy đại đa số các từ có chứa kí tự không thuộc phạm vi ASCII. Bởi vậy ta lựa chọn tiêu chí là từ hạt giống phải có ít nhất 1 kí tự Unicode không thuộc phạm vi ASCII, các từ khác sẽ không được xét, các chữ số hoặc các mục không phải kí tự cũng sẽ bị loại trừ. Ở đây, ta bỏ qua 1000 từ có tần suất cao nhất vì chúng thường được coi là các từ dừng (*stop word*) đối với các máy tìm kiếm. 5000 từ tiếp theo trong danh sách tần suất thuộc nhóm từ có tần suất trung bình được sử dụng làm từ hạt giống.

3.2. Thu thập dữ liệu từ Web

Việc thu thập dữ liệu từ Web được thực hiện bằng cách lặp lại nhiều nghìn lần cho đến khi thu được kho ngữ liệu đủ lớn:

- Lựa chọn ngẫu nhiên một số từ trong số các từ hạt giống để tạo nên một truy vấn.
- Gửi truy vấn tới một máy tìm kiếm (như Google hay Yahoo).
- Tải về tất cả các tài liệu kết quả của máy tìm kiếm và lưu lại.

3.2.1. Sinh truy vấn

Các truy vấn Web được sinh ra từ tập các từ hạt giống bằng cách sử dụng thành phần sinh truy vấn của công cụ BooTCaT [12]. Thành phần này sinh ra các truy vấn có độ dài n bằng cách rút ngẫu nhiên n từ. Các bộ n từ không giống hệt nhau và cũng không là hoán vị của nhau. Ta phải xác định độ dài hợp lý của truy vấn để xác suất kết quả tìm kiếm thuộc đúng ngôn ngữ cần tìm là cao, đồng thời phải đảm bảo số lượng các URL tìm được là không nhỏ đối với hầu hết các truy vấn. Chừng nào số lượng URL tìm được lớn hơn 10 cho hầu hết các truy vấn (chẳng hạn 90 %) thì độ dài của truy vấn được coi là hợp lệ. Ở đây, ta định nghĩa độ dài truy vấn tốt nhất là độ dài tối đa của một truy vấn mà trong đó số lượng kết quả được tìm ra hầu hết là lớn hơn 10. Thuật toán sau được sử dụng để xác định độ dài tốt nhất cho mỗi truy vấn:

1. Đặt $n = 1$
2. Sinh ra 100 truy vấn, mỗi truy vấn có độ dài bằng n

3. Sắp xếp các truy vấn theo số các kết quả tìm được

4. Đếm số kết quả tìm được ở truy vấn thứ 90 (*min-hits-count*)

5. Nếu *min-hits-count* <10 thì dừng thuật toán và trả về $n-1$

6. $n = n + 1$; Quay lại bước 2

Độ dài truy vấn tốt nhất cho tiếng Việt khi tìm kiếm trên Yahoo được chỉ ra ở Bảng 1.

Bảng 1. Độ dài truy vấn, số trang kết quả ở truy vấn thứ 90, độ dài tốt nhất.

$n = 1$	2	3	4	5	Độ dài tốt nhất
1.100.000	15.400	422	39	5	4

Sau khi xác định độ dài truy vấn, sinh ra khoảng 30.000 truy vấn.

3.2.2. Thu thập địa chỉ URL (Uniform Resource Locator)

Sử dụng hàm API (*Application Programming Interface*) của Yahoo để thực hiện tìm kiếm đối với 30.000 truy vấn, mỗi truy vấn thu lấy mười kết quả tìm kiếm đầu tiên. Nếu một URL xuất hiện nhiều lần thì chỉ giữ lại duy nhất một URL. Số liệu thống kê thu thập URL và lọc dữ liệu được trình bày ở Bảng 2.

Bảng 2. Thống kê kho dữ liệu từ Web

Số lượng các URL thu được	Số lượng sau khi lọc	Số lượng sau khi loại bỏ phần trùng lặp gần nhau	Dung lượng kho dữ liệu thu được trên Web	
			MB	Từ
106.076	27.728	19.646	1200 GB	149 triệu từ

Thành phần thu thập URL của BooTCaT được mở để lưu trữ truy vấn hiện tại, kích thước trang và kiểu MIME (*Multipurpose Internet Mail Extensions*) cho mỗi URL.

3.3. Lọc ngữ liệu

Khi các URL được tải về, thông tin MIME cho URL cũng như kích cỡ của trang là có sẵn, chỉ thu lấy các trang có kiểu MIME là *text* hoặc HTML và có dung lượng lớn hơn 5 KB (để xác suất các tệp này chứa văn bản liên quan là lớn hơn). Các tệp có dung lượng lớn hơn 2 MB cũng được loại bỏ để tránh bất kì tệp thuộc miền đặc biệt nào thống trị thành phần của kho ngữ liệu, và cũng bởi vì các tập tin có độ lớn này rất nhiều trường hợp là các tệp nhật kí hay là các văn bản không liên quan khác.

Các trang web được tải về bao hàm cả các thẻ HTML, các thành phần văn bản *boilerplate* kiểu như thanh duyệt nội dung, quảng cáo, v.v. Để loại bỏ các nội dung như vậy và chỉ giữ lại phần văn bản liên quan, ta sử dụng thuật toán BTE *(Body Text Extraction)* [2]. BTE bắt

nguồn từ quan sát là các trang Web thường có phần đầu và phần cuối chứa nhiều *boilerplate* và thẻ HTML, còn phần thân văn bản ở giữa có chứa ít thẻ chính là phần tài liệu ta quan tâm. Thuật toán BTE tính toán tỉ lệ phần văn bản để đánh dấu cho các phần khác nhau của trang, chia trang thành ba phần trên cơ sở tỉ lệ này và chỉ giữ lại một phần giữa. BTE đã được thực hiện trên tất cả các trang tải xuống để thu được các trang văn bản thô.

Những trang văn bản thô này lại tiếp tục được kiểm tra tính kết nối văn bản - văn bản kết nối trong các câu phải chứa một tỉ lệ các từ chức năng cao [10]. Nếu một trang không đáp ứng tiêu chí này thì sẽ bị loại. Việc kiểm tra thực hiện như sau. Ta giả định 500 từ đầu tiên trong danh sách tần suất từ (có được nhờ kho ngữ liệu Wiki) chứa hầu hết các từ chức năng. Để thiết lập một ngưỡng cho tỉ lệ các từ chức năng trong văn bản kết nối, chúng tôi sắp xếp tất cả các tệp tin Wiki theo tỉ lệ các từ thuộc 500 từ đầu tiên này trong mỗi tệp. Ta thấy rằng hầu hết các tệp Wiki ở phía cuối danh sách đã được sắp xếp này (sau khoảng 75-80%) không chứa văn bản kết nối. Đây hoặc là do công cụ *Wikipedia2Text* làm sạch không tốt hoặc vì tệp thực sự không có kết nối văn bản. Tệp tin Wiki ở vị trí thứ 70% của danh sách đã sắp xếp được sử dụng để thiết lập ngưỡng: Nếu danh sách 500 từ đầu chiếm 65% của tất cả các từ trong tệp thứ 70% thì ngưỡng của ngôn ngữ được đặt bằng 65%. Khi đó bất kỳ trang nào có ít hơn 65% số từ thuộc 500 từ đầu tiên trong danh sách tần suất từ sẽ bị loại bỏ.

3.4. Phát hiện tài liệu gần trùng lặp

Sử dụng mô-đun Text::DeDuper viết bằng Perl để phát hiện tài liệu gần trùng lặp nhau[4]. Mô-đun này sử dụng độ đo độ giống nhau như đề xuất của Broder và cs [22] để phát hiện các tài liệu tương tự nhau dựa vào văn bản trong đó. Đây là nhiệm vụ cần nhiều bộ nhớ: cần sinh mô hình n-gram ($n = 5$) cho mỗi tài liệu và đo độ tương tự (ngưỡng = 0.2) giữa 2 tài liệu dựa trên số n-gram giao nhau của chúng. Do kích thước bộ nhớ trong là hạn chế và chỉ có thể chứa một số hữu hạn các tệp, nên việc phát hiện trùng lặp được thực hiện bằng cách tiếp cận dùng cửa sổ trượt. Trước tiên, tất cả các tệp sẽ được sắp xếp theo kích cỡ và lưu tên tệp vào một danh sách. Mỗi lần lặp, mô-đun DeDuper xác định được một số cố định (500) các tệp không trùng nhau trong danh sách (duyệt tuần tự trên các tệp đã được sắp xếp theo kích cỡ) mà mô hình n-gram của chúng có thể vừa với bộ nhớ. Với tất cả các tệp còn lại trong danh sách, so sánh từng tệp với các n-gram của các tệp không trùng lặp trên để xác định chúng có trùng lặp hay không, nếu có thì loại bỏ. Quá trình này được lặp cho tới khi tất cả các tệp được xử lí.

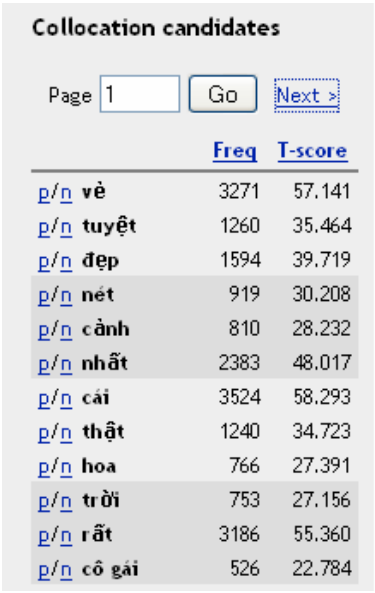
3.5. Tách từ và gán nhãn từ loại

Để thu được kho ngữ liệu tiếng Việt mà trong đó các văn bản đã được tách từ và gán nhãn từ loại, công cụ tách từ vnTokenizer [16] và công cụ gán nhãn từ loại vnTagger [15] được sử dụng. Các công cụ này do nhóm nghiên cứu xây dựng, sử dụng dữ liệu huấn luyện là kho treebank tiếng Việt [19].

Sau khi tách từ và gán nhãn từ loại, kho ngữ liệu được tích hợp vào hệ thống Sketch Engine. Hình 3.1 minh họa cho việc sử dụng chức năng *Concordance* trong hệ thống để khai thác kho ngữ liệu tiếng Việt phục vụ thống kê tần suất và tính trội của các từ lân cận với một từ bất kỳ. Trong đó, tính trội được thống kê theo tỷ lệ của việc quan sát thực tế với giả thiết

[4]Trong tương lai, chúng tôi sẽ khai thác và sử dụng phương pháp của Pomikálek và Rychlý [9]

đảo (của các từ lân cận cùng xuất hiện với một từ bất kỳ) thông qua công thức T-score hoặc MI-score [5].



Hình 3.1. Danh sách tần suất và tính trội của các từ lân cận với tính từ "đẹp"

4. XÂY DỰNG TẬP QUAN HỆ NGỮ PHÁP TIẾNG VIỆT

Tiếp theo việc xây dựng kho ngữ liệu, ta xây dựng tập quan hệ ngữ pháp tiếng Việt phục vụ cho chức năng Word Sketch của hệ thống. Ngôn ngữ truy vấn kho ngữ liệu (CQL) được sử dụng để biểu diễn các quan hệ ngữ pháp dùng cho truy vấn kho ngữ liệu. Bởi vậy, trước hết sẽ giới thiệu về ngôn ngữ truy vấn kho ngữ liệu, sau đó sẽ trình bày về việc xây dựng bộ quan hệ ngữ pháp.

4.1. Ngôn ngữ truy vấn kho ngữ liệu

Ngôn ngữ truy vấn kho ngữ liệu [6] sử dụng trong Sketch Engine được phát triển bởi nhóm từ vựng và kho ngữ liệu tại IMS, Trường Đại học Stuttgart vào những năm 1990. Mỗi truy vấn là một biểu thức chính quy trên các biểu thức thuộc tính (ví dụ thuộc tính *word* cho từ và *tag* cho nhãn từ loại). Một biểu thức kiểm tra trong truy vấn có khuôn dạng: *attribute_nameoperatorstring*. ở đây, *attribute_name* là tên thuộc tính, *operator* là toán tử phù hợp (=) hoặc phủ định (! =), *string* là một xâu cụ thể hoặc một biểu thức chính quy.

Các ví dụ sau minh họa một số truy vấn theo ngôn ngữ CQL.

1) Tìm kiếm các từ bắt đầu với confuse, với nhiều nhất 10 từ nằm giữa, sau cùng là giới từ hoặc danh từ chỉ người:

[5]http://trac.sketchengine.co.uk/wiki/SkE/DocsIndex
[6]http://www.fi.muni.cz/ thomas/corpora/CQL/

"confuse.*" []{0,10} [tag="IN" | tag="PP"]

2) Tìm kiếm một chuỗi gồm có một tính từ, một danh từ, một từ kết nối và một danh từ khác:

[tag="JJ.*"][tag="N.*"]"and|or" [tag="N.*"]

4.2. Xây dựng quan hệ ngữ pháp

Để xác định quan hệ ngữ pháp giữa các từ, Sketch Engine cần biết làm thế nào tìm được các từ kết nối với nhau theo một quan hệ ngữ pháp trong ngôn ngữ đang xét. Sketch Engine cho phép làm việc này theo 2 cách.

1. Kho ngữ liệu đầu vào đã được phân tích cú pháp và thông tin về từ nào có quan hệ ngữ pháp nào với các từ khác đã được nhúng trong kho ngữ liệu. Hiện tại, các kho ngữ liệu có chú giải cú pháp phụ thuộc đã hoàn toàn được hỗ trợ. Các cây cú pháp thành phần cần có thành phần chính của các ngữ đoạn được đánh dấu.

2. Kho ngữ liệu đầu vào chưa được phân tích cú pháp, và Sketch Engine hỗ trợ quá trình xác định các thành phần quan hệ ngữ pháp. Trong trường hợp này, các chuyên gia sẽ định nghĩa từng quan hệ ngữ pháp [7], sử dụng Sketch Engine để kiểm tra và phát triển nó, và cuối cùng đưa bộ quan hệ ngữ pháp vào hệ thống Sketch Engine. Sau đó thì người sử dụng sẽ có thể sử dụng Sketch Engine để tìm được tất cả các từ có quan hệ ngữ pháp thông qua chức năng Word Sketch.

Đối với tiếng Việt, ta chưa có kho ngữ liệu lớn đã phân tích cú pháp nên phải sử dụng khả năng thứ hai, tức là cần định nghĩa được tập các quan hệ ngữ pháp.

Hệ hình thức dùng cho các quan hệ ngữ pháp ở đây là dựa trên cơ sở các mẫu xâu theo biểu thức chính quy, do đó phù hợp với các ngôn ngữ có trật tự từ ổn định, như tiếng Anh chẳng hạn. Tiếng Việt cũng là một ngôn ngữ trong đó trật tự từ đóng vai trò quan trọng.

Ví dụ về định nghĩa một quan hệ ngữ pháp: Muốn định nghĩa quan hệ "động từ - bổ ngữ", ta thấy rằng bổ ngữ của động từ có thể là một cụm danh từ, cụm động từ, cụm giới từ... Xét trường hợp bổ ngữ là cụm danh từ, trong đó từ trung tâm là một danh từ và cũng là danh từ cuối cùng của một chuỗi bao gồm các số từ (M), tính từ (A), trạng từ (R) và các danh từ khác (N), và chú ý rằng đối với mỗi cụm danh từ bổ ngữ cho động từ (V), theo mặc định cụm danh từ này luôn đứng trực tiếp sau động từ trong câu, nếu động từ đang xét nằm trong một nhóm các động từ thì nó thường là động từ cuối cùng trong nhóm đó. Những thông tin này cho ta một định nghĩa về quan hệ "động từ-bổ ngữ" là: quan hệ "một chuỗi bắt đầu với động từ và cuối cùng là danh từ, ở giữa là một chuỗi bất kỳ của các trạng từ hoặc các định từ hoặc các số từ hoặc các tính từ hoặc các danh từ khác". Khi đó, ta xây dựng được một biểu diễn cho mẫu ngữ pháp trên:

1:"V" "(M|A|R|N)"* 2:"N"

Ở đây, nhãn 1: và 2: đánh dấu việc rút trích các từ của đối thứ nhất và thứ hai trong mối quan hệ ngữ pháp, các phép toán | và phép * là phép toán trong biểu thức chính quy.

[7] http://trac.sketchengine.co.uk/wiki/SkE/CorpusQuerying

Nhóm nghiên cứu đã xây dựng được tệp quan hệ ngữ pháp cơ bản tiếng Việt phiên bản 1 [8] phục vụ cho hệ thống Sketch Engine. Trong đó, mỗi quan hệ ngữ pháp cơ bản sẽ bao gồm một số mệnh đề, mỗi mệnh đề được viết trên một dòng bằng ngôn ngữ CQL thông qua biểu thức chính quy trên thuộc tính nhãn từ loại. Các quan hệ này được xây dựng dựa vào các tài liệu ngữ pháp tiếng Việt [20].

Dưới đây là một số ví dụ định nghĩa quan hệ ngữ pháp mà nhóm nghiên cứu đã xây dựng.
Ví dụ 1:

*DUAL

$= A\_modifies\_N/N\_modifier\_A$

1:"N" "P|R|A" {0,3} 2:"A"

Từ khóa DUAL để xác định là có 2 mối quan hệ được định nghĩa ở đây. $A\_modifies\_N$ - danh từ được bổ nghĩa bởi các tính từ và phần đối của nó $N\_modifier\_A$ – tính từ bổ nghĩa cho các danh từ. Chuỗi kí tự sau dấu bằng là các tên của các quan hệ, được cách nhau bởi dấu (/). Sau cùng là biểu thức chính qui biểu diễn mối quan hệ ngữ pháp giữa tính từ và danh từ.
Ví dụ 2:

*SYMMETRIC

$= conjunction$

1:"N.?[word="và"|word="hoặc"|word=","]{1} 2:"N.?"

Từ khoá SYMMETRIC để xác định mối quan hệ là đối xứng. Quan hệ được định nghĩa có tên là *conjunction*, xác định mối quan hệ kết hợp giữa hai danh từ thông qua liên từ "hoặc", "và", và ",".

Ta đã tích hợp kho ngữ liệu tiếng Việt đã tách từ và gán nhãn từ loại tự động và tập quan hệ ngữ pháp vào hệ thống Sketch Engine. Kho ngữ liệu chứa khoảng 100 triệu từ, còn tập quan hệ ngữ pháp gồm 11 bộ quan hệ chính. Việc tích hợp này cho phép người sử dụng có thể thực hiện mọi chức năng của Sketch Engine phục vụ cho việc nghiên cứu từ vựng tiếng Việt.

Hình 4.1 minh họa cho chức năng Word Sketch của hệ thống Sketch Engine. Trong đó, mỗi bảng là một danh sách thống kê tần suất và tính trội của các từ trong cùng mối quan hệ ngữ pháp với 1 từ bất kỳ (ở đây là tính từ "đẹp").

Bảng $R\_Modfies\_A$: Danh sách các từ bổ nghĩa cho tính từ "đẹp"

Bảng $N\_Modifier\_A$: Danh sách các danh từ có tính từ "đẹp" là từ bổ nghĩa

Bảng *Conjnuntion*: Danh sách các tính từ kết hợp với tính từ "đẹp" thông qua các liên từ

Bảng *AdjAdverb*: Danh sách các từ mà "đẹp" là phó từ của các từ đó

5. KẾT LUẬN

Bài báo đã giới thiệu hệ thống Sketch Engine và nghiên cứu triển khai hệ thống này cho tiếng Việt. Đề xuất cách thức xây dựng kho ngữ liệu và tập các quan hệ ngữ pháp cơ bản tiếng Việt để phục vụ cho hệ thống truy vấn kho ngữ liệu trong Sketch Engine. Hiện tại việc đánh giá chất lượng của bộ quan hệ ngữ pháp đang được thực hiện thông qua người dùng (Trung tâm từ điển VietLex). Trong thời gian tới nhóm nghiên cứu sẽ tiếp tục đánh giá và

[8] $http://the.sketchengine.co.uk/auth/sketch_grammar/1353/view/$

đẹp (-j) VietnameseWaCTagged freq = 32389 (249.6 per million)

R_modifies_A	6053	20.5	N_modifier_A	20673	19.0	AdjAdVerb	3243	6.4	conjunction	1845	1.6
tuyệt	292	9.77	vẻ	3076	11.22	ăn mặc	74	8.34	thơ mộng	39	8.82
thật	529	8.07	nét	747	9.19	trang trí	46	7.55	lãng mạn	46	8.13
rất	2251	7.69	Vẻ	254	8.59	trông	156	7.49	duyên dáng	19	7.8
khá	227	6.96	cảnh	587	8.21	Sống	31	7.47	sang trọng	30	7.78
càng	110	5.72	cô gái	360	8.14	vẽ	85	6.87	nên thơ	11	7.3
vừa	151	5.63	gái	213	7.78	mặc	107	6.71	độc đáo	26	7.02
quá	178	5.54	hình ảnh	328	7.59	Trông	14	6.69	gợi cảm	10	6.98
quả là	9	5.08	kỷ niệm	164	7.33	khen	35	6.59	hoành tráng	14	6.9
Thật	11	4.94	phong cảnh	107	7.27	trang hoàng	11	6.37	dễ thương	15	6.62
cực	14	4.86	Nét	101	7.23	thắng	91	6.36	lịch sự	12	6.56
vẫn	174	4.86	đàn bà	161	7.19	thiết kế	43	6.12	trong sáng	14	6.52
cũng	423	4.8	hoa	297	7.03	múa	18	6.09	trang nhã	6	6.52
Không chỉ	6	4.71	ảnh	194	6.95	nở	25	5.94	yên tĩnh	10	6.46
quả thật	7	4.71	giấc mơ	96	6.87	chạm trổ	7	5.83	trang nghiêm	10	6.44
vô cùng	18	4.57	con gái	155	6.86	tạo hình	7	5.68	nhỏ nhắn	6	6.4
Rất	10	4.55	nắng	115	6.8	chạm khắc	6	5.66	khang trang	6	6.27
cực kỳ	9	4.5	quần áo	103	6.77	chuyển	9	5.6	lung linh	6	6.15
có lẽ	10	4.22	Ảnh	90	6.74	Quá	7	5.58	sạch sẽ	8	6.1

Hình 4.1. Các bảng danh sách từ có quan hệ ngữ pháp với tính từ "đẹp".

nâng cao chất lượng của bộ quan hệ ngữ pháp. Đồng thời sẽ tiếp cận khả năng xây dựng tập quan hệ ngữ pháp cơ bản bằng cách rút trích tự động các quan hệ ngữ pháp từ kho ngữ liệu đã được chú giải cú pháp tiếng Việt để có độ phủ rộng hơn.

TÀI LIỆU THAM KHẢO

[1] A. Ferraresi, E. Zanchetta, M. Baroni, and S. Bernardini, Introducing and evaluating "ukwac", a very large web-derived corpus of english, *Proceedings of the 4th Web As Corpus Workshop at LREC,* Marrakech, Morocco, 2008.

[2] A. Finn, N. Kushmerick, and B. Smyth, Fact or fiction: Content classification for digital libraries, *Proceedings of the Second DELOS Network of Excellence Workshop on Personalisation and Recommender Systems in Digital Libraries,* Dublin City University, Ireland, 2001.

[3] A. Kilgarriff, Simple maths for keywords, *Proceedings of the Corpus Linguistics Conference,* University of Liverpool, UK, 2009.

[4] A. Kilgarriff and M. Rundell, Lexical profiling software and its lexicographic applications: a case study, *Proceedings of EURALEX,* Copenhagen, 2002(807–818).

[5] A. Kilgarriff, P. Rychlý, P. Smrz, and D. Tugwell, The sketch engine, *Proceedings of EURALEX,* Lorient, France (http://www.sketchengine.co.uk/), 2004.

[6] A. Kilgarriff, S. Reddy, J. Pomikálek, and Avinesh PVS, A corpus factory for many languages, *Proceedings of the Seventh conference on International Language Resources and Evaluation,* (LREC'10) (Valletta, Malta) (Nicoletta Calzolari(̃Conference Chair), Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, eds.), European Language Resources Association (ELRA), 2010.

[7] F. Keller and M. Lapata, Using the web to obtain frequencies for unseen bigrams, *Computational Linguistics*, **29**(2)(2003)459–484.

[8] G. Grefenstette and J. Nioche, Estimation of english and non-english language use on the www, *Proceedings of RIAO, Recherche d'Informations Assistée par Ordinateur*, Paris, 2000(237—246).

[9] J. Pomikálek and P. Rychlý, Detecting co-derivative documents in large text collections, *Proceedings of the Sixth Conference on International Language Resources and Evaluation, LREC'08*, Marrakech, Morocco, 2008.

[10] M. Baroni, *Distributions In Text*, Corpus Linguistics: An International Handbook, Anke Lüdeling and Merja Kytö, eds., vol. 2, Mouton de Gruyter, Berlin, 2007(803–821).

[11] M. Baroni and A. Kilgarriff, Large linguistically-processed web corpora for multiple languages, *Proceedings of EACL*, 2006(87–90).

[12] M. Baroni and S. Bernardini, Bootcat: Bootstrapping corpora and terms from the web, *Proceedings of LREC 2004*, Lisbon, Portugal, 2004(1313—1316).

[13] M. J. Sinclair, *Looking up: an account of the cobuild project in lexical computiting*, Collins, 1987.

[14] N. Ide, R. Reppen, and K. Suderman, The american national corpus: more than the web can provide, *Proceedings of the Third Language Resources and Evaluation Conference,* Las Palmas, 2002(839–844).

[15] P. Le Hong, A. Roussanaly, T. M. H. Nguyen, and M. Rossignol, An empirical study of maximum entropy approach for part-of-speech tagging of vietnamese texts, *Proceedings of the 37^{th} annual meeting of the Association for Computational Linguistics on Computational Linguistics, TALN*, Montréal, 2010.

[16] P. Le Hong, T.M.H. Nguyen, A. Roussanaly, and T.V. Ho, A hybrid approach to word segmentation of vietnamese texts, *Proceedings of the 2^{th} international conference of the Language and Automata Theory and Applications,* (Tarragona, Spain), vol. 5196, Springer Berlin, 2008 (240–249).

[17] R. Jones and R. Ghani, Building a corpus for a minority language from the web, *Proceedings of the Student Workshop of the 38^{th} Annual Meeting of the Association for Computational Linguistics*, 2000 (29-–36).

[18] S. Sharoff, Creating general-purpose corpora using automated search engine queries, *WaCky! Working papers on the Web as Corpus* (Marco Baroni and Silvia Bernardini, eds.), Gedit, Bologna, 2006.

[19] P.T. Nguyen, X.L. Vu, T.M.H. Nguyen, V.H. Nguyen, and P. Le Hong, Building a large syntactically-annotated corpus of vietnamese, *Proceedings of the 3^{th} Linguistic Annotation Workshop, ACL-IJCNLP*, Singapore, 2009.

[20] Nguyễn Minh Thuyết and Nguyễn Văn Hiệp, *Thành phần câu tiếng Việt*, NXB Đại học Quốc gia Hà Nội, 1998.

[21] K.W. Church and P. Hanks, Word association norms, mutual information and lexicograph, *Proceedings of 27^{th} Annual Meeting of ACL,* Vancouver, 1989 (76–83).

[22] A.Z. Broder, S.C. Glassman, M.S. Manasse, and G. Zweig, Syntactic clustering of the web, *Computer Networks* **29** (8-13)(1997) 1157–1166.

Ngày nhận bài 9 - 7 - 2011

Nhận lại sau sửa 13 - 9 - 2011