*Tạp chí Tin học và Điều khiển học, T.23, S.3 (2007), 272–283*

# DỊCH TỰ ĐỘNG TRUY VẤN TIẾNG VIỆT SANG ĐỒ THỊ KHÁI NIỆM

HỒNG TRUNG DŨNG, CAO HOÀNG TRỤ

*Khoa Khoa học và Kỹ thuật Máy tính, Trường Đại học Bách Khoa Tp. Hồ Chí Minh*

**Abstract.** Current keyword-based systems like Google are very helpful, yet neither sound nor complete, because keywords are not adequate to represent the semantics of documents and queries. Searching based on semantics in general, and named entities in particular, would enable more intelligent searching services. While there are various formalisms to represent queries with precise semantics, natural language is still the most desirable means to users. On the basis of the Vietnamese Semantic Web system VN-KIM, which can recognize named entities in Vietnamese documents, this paper proposes a method to translate a query written in Vietnamese to a corresponding conceptual graph, for searching the documents containing the entities specified in the query. Implementation and experiment results are also presented and evaluated.

**Tóm tắt.** Các hệ thống tìm kiếm theo từ khoá hiện nay như Google rất hữu ích, nhưng chưa chính xác và chưa đầy đủ, do các từ khoá chưa biểu diễn được hết ngữ nghĩa của các tài liệu và truy vấn. Tìm kiếm theo ngữ nghĩa nói chung, và thực thể có tên nói riêng, sẽ cho phép các dịch vụ tìm kiếm thông minh hơn. Trong khi có nhiều ngôn ngữ hình thức để biểu diễn truy vấn với ngữ nghĩa chính xác, ngôn ngữ tự nhiên vẫn là phương tiện mà người sử dụng mong muốn nhất. Dựa trên hệ thống Web Việt có Ngữ nghĩa VN-KIM, với khả năng nhận diện được các thực thể có tên trong các tài liệu tiếng Việt, bài báo này đề xuất một giải pháp dịch tự động một câu truy vấn tiếng Việt sang một đồ thị khái niệm tương ứng, để tìm kiếm các tài liệu chứa các thực thể có tên nêu trong câu truy vấn. Kết quả hiện thực và thí nghiệm cũng được trình bày và đánh giá.

## 1. GIỚI THIỆU

World Wide Web (gọi tắt là Web) đã trở thành một kho tàng thông tin khổng lồ của nhân loại và một môi trường chuyển tải thông tin không thể thiếu được trong thời đại ngày nay. Các hệ thống tìm kiếm theo từ khóa hiện nay như Google đã bộc lộ khuyết điểm về tính chính xác và tính đầy đủ, do các từ khóa không biểu diễn được hết ngữ nghĩa của các tài liệu cũng như truy vấn. Ví dụ, với truy vấn tìm các trang Web nói về thành phố Sài Gòn, người sử dụng mong đợi nhận được các trang đề cập đến Sài Gòn hoặc TP.HCM như một thành phố. Trong khi đó, một hệ thống tìm kiếm theo từ khoá có thể trả về các trang Web có chứa từ "thành phố", mặc dù nội dung nói về các thành phố khác, hoặc chứa từ "Sài Gòn", mặc dù nội dung nói về sông hoặc các thực thể khác mang tên Sài Gòn.

Vì vậy, ý tưởng và mục tiêu của Web có Ngữ nghĩa ([2]), thế hệ sắp tới của Web, là làm cho các trang Web có được ngữ nghĩa mà máy tính có thể hiểu và xử lý tự động. Trong một tài liệu, các thực thể có tên (Named Entity), là những thực thể có thể được tham khảo đến

bằng tên, như con người, tổ chức, nơi chốn, tạo nên phần cơ bản trong ngữ nghĩa của tài liệu đó. Nhiều hệ thống nhận diện tự động các thực thể có tên đã và đang được phát triển, trong đó KIM là một hệ thống được phát triển một cách bài bản và đạt được những kết quả đáng chú ý nhất trên các tài liệu và miền tri thức tiếng Anh ([10]). Một hệ thống tương tự cho tiếng Việt, VN-KIM, cũng vừa được hoàn thành ([3]). Khi các trang Web đã được chú thích ngữ nghĩa về các thực thể có tên, các dịch vụ tìm kiếm cũng có thể trở nên thông minh hơn, với kết quả trả về chính xác và đầy đủ hơn đối với các văn bản thô chỉ dựa trên các từ khoá. Trong [6], các tác giả mô tả một hệ thống tìm kiếm theo thực thể có tên như vậy, bằng cách bổ sung thêm dữ liệu cho các tên riêng trong một truy vấn trước khi tiến hành tìm kiếm. Ví dụ, với tên "Washington" trong một truy vấn, cần xác định đó là tên người hay nơi chốn để tìm thấy các tài liệu phù hợp. Tuy nhiên, các tác giả chưa trình bày được một giải pháp cụ thể cho việc này. Trong [4], đồ thị khái niệm (Conceptual Graph) ([14]) được đề xuất làm ngôn ngữ để đặc tả các thực thể có tên trong truy vấn tìm các tài liệu liên quan đến chúng.

Đồ thị khái niệm là một ngôn ngữ biểu diễn tri thức trực quan, có thể chuyển đổi qua lại tương đối dễ dàng với ngôn ngữ tự nhiên ([15]) và ngôn ngữ biểu diễn tri thức trên Web phổ biến hiện nay là RDF ([17]). Tuy đồ thị khái niệm thân thiện với người sử dụng hơn so với các ngôn ngữ hình thức khác, ngôn ngữ tự nhiên vẫn luôn là mong muốn nhất. Vì vậy, trọng tâm của nghiên cứu này là phát triển một hệ thống tìm kiếm thông tin theo thực thể có tên và cho phép các truy vấn bằng tiếng Việt. Theo đó, một truy vấn tiếng Việt được dịch sang một đồ thị khái niệm tương ứng về ngữ nghĩa, trước khi tiến hành tìm kiếm.

Bài toán rút trích ngữ nghĩa đầy đủ của một văn bản là cực khó, bao gồm việc nhận diện các thực thể có tên và quan hệ giữa chúng trong văn bản đó. Các dự án như S-CREAM ([7]) và MnM ([16]) sử dụng rất nhiều các kỹ thuật học máy để rút trích các quan hệ giữa các thực thể, tuy nhiên chỉ làm được một cách bán tự động. Trong [18], các tác giả đề xuất một phương pháp học hoàn toàn tự động, sử dụng văn phạm liên kết và chuyển đổi các câu ở văn phạm này sang đồ thị RDF hoặc đồ thị khái niệm, nhưng độ chính xác đạt được chỉ khoảng 60% và chỉ áp dụng cho một miền hẹp cụ thể. Một kết quả tương tự đạt được trong [8], dựa trên các từ loại và mẫu câu trong văn bản.

Do đó, trong phạm vi của nghiên cứu này, chúng tôi chỉ tập trung vào bài toán dịch các cụm từ tiếng Việt biểu diễn truy vấn sang đồ thị khái niệm, nhằm tránh độ phức tạp quá lớn về cú pháp và ngữ nghĩa của các câu đầy đủ. Có hai nghiên cứu gần với chúng tôi về việc sử dụng ngôn ngữ tự nhiên cho truy vấn và một ngôn ngữ hình thức để biểu diễn ngữ nghĩa chính xác là [1] và [13]. Trong [1], tác giả sử dụng Logic mô tả để biểu diễn ngữ nghĩa của truy vấn và phát triển một giải thuật để dịch một truy vấn bằng Logic mô tả sang SQL. Tuy nhiên, tác giả chưa đưa ra được một lời giải cụ thể cho bài toán dịch từ một truy vấn ngôn ngữ tự nhiên sang Logic mô tả. Trong [13], các tác giả phác thảo một hệ thống truy hồi thông tin trong đó ngữ nghĩa của các tài liệu được biểu diễn bằng đồ thị khái niệm và các truy vấn bằng ngôn ngữ tự nhiên, nhưng chưa có giải pháp để sinh đồ thị khái niệm cho các tài liệu cũng như truy vấn.

Trong hệ thống mà chúng tôi đề xuất, sau khi một truy vấn tiếng Việt đã được dịch sang đồ thị khái niệm, khối tiếp nhận và xử lý truy vấn ở dạng đồ thị khái niệm của VN-KIM hiện thời được tận dụng lại để tìm kiếm và trả về kết quả. Mục 2 tiếp theo giới thiệu tóm tắt về VN-KIM và đồ thị khái niệm. Các Mục 3 và 4 trình bày các bước sinh đồ thị khái niệm từ

một truy vấn tiếng Việt. Mục 5 mô tả các thí nghiệm để đánh giá hiệu quả biên dịch. Cuối cùng, Mục 6 nêu một số kết luận và hướng nghiên cứu tiếp theo.

## 2. VN-KIM VÀ ĐỒ THỊ KHÁI NIỆM

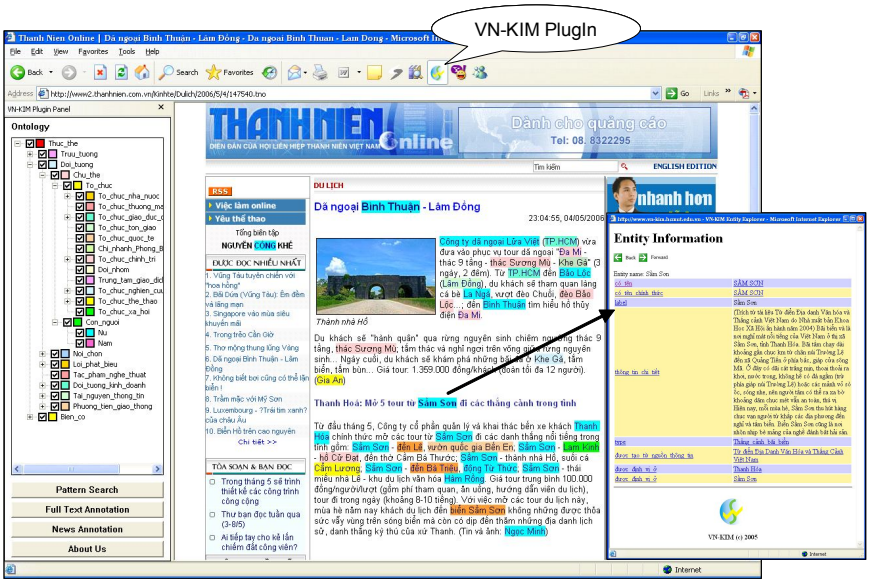
Hệ thống VN-KIM được xây dựng để làm một cơ sở hạ tầng đầu tiên về web Việt có ngữ nghĩa, bao gồm một Ontology và cơ sở tri thức về các thực thể có tên phổ biến nhất ở Việt Nam, và các phần mềm căn bản để rút trích và truy hồi thông tin theo thực thể có tên. VN-KIM bao gồm các khối chính sau:

1. Ontology và cơ sở tri thức: gồm một cây phân cấp các lớp và tri thức về các thực thể có tên phổ biến ở Việt Nam và trên thế giới. Hiện tại, VN-KIM Ontology có 370 lớp với 36 thuộc tính và 79 quan hệ, và cơ sở tri thức có trên 120.000 thực thể.

2. Phần mềm rút trích thông tin và chú thích ngữ nghĩa: nhận vào một trang Web, phân tích trang Web để rút trích ra các khối văn bản mang tin tức chính, và sau đó nhận diện các thực thể có tên và chú thích ngữ nghĩa của chúng vào trang Web. Giá trị trung bình điều hoà (F-Measure) của độ chính xác và độ đầy đủ mà VN-KIM hiện đạt được là khoảng 80% ([11]).

3. Phần mềm truy hồi thông tin theo thực thể có tên: dùng để soạn thảo và thực hiện truy vấn trên cơ sở tri thức và kho các trang Web có chú thích ngữ nghĩa của VN-KIM. Phần mềm này hiện hỗ trợ người dùng ba cách thức truy vấn là dùng ngôn ngữ truy vấn SeRQL của Sesame ([9]), các mẫu câu truy vấn đã được thiết lập sẵn, hoặc đồ thị khái niệm.

4. Phần mềm xây dựng và phát triển cơ sở tri thức: dùng để xây dựng và cập nhật một cơ sở tri thức lưu trữ dưới dạng RDF. Phần mềm này có một tính năng nổi trội hơn Protégé ([12]) là cho phép xây dựng và cập nhật cục bộ và không trực tuyến từng phần cơ sở tri thức, nhờ đó có thể phân tải việc nhập tri thức và quản trị được một cơ sở tri thức lớn.

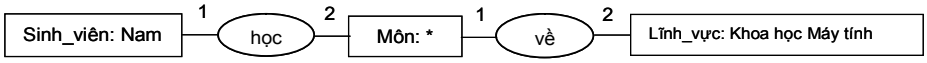



*Hình 1.* Kết quả nhận diện thực thể của VN-KIM

Hình 1 cho thấy giao diện của VN-KIM PlugIn và kết quả nhận diện trực tuyến các thực

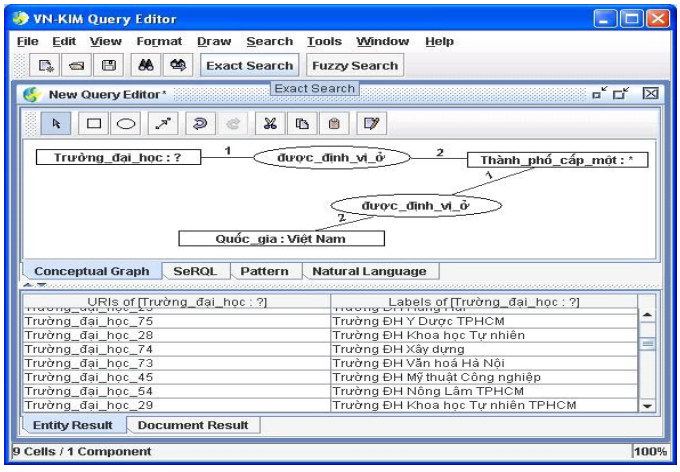
thể trên một trang Web của báo Thanh Niên. Cột bên trái thể hiện Ontology của VN-KIM và các nút chức năng chú thích và tìm kiếm tài liệu theo thực thể có tên; bên phải là nội dung trang Web đã được chú giải, trong đó các thực thể được tô các màu tương ứng với lớp của chúng trong Ontology. Người đọc tin có thể duyệt thông tin chi tiết của một thực thể, nếu có trong cơ sở tri thức của VN-KIM, bằng cách nhấp chuột vào thực thể đó và đi theo các siêu liên kết trên trình duyệt thực thể của VN-KIM.

Để truy vấn tri thức hoặc tài liệu về một thực thể có tên, VN-KIM cho phép đặt câu truy vấn bằng đồ thị khái niệm. Một đồ thị khái niệm là một đồ thị lưỡng phân với các nút khái niệm xen kẽ với các nút quan hệ, nối với nhau bằng các cạnh. Mỗi nút khái niệm, được vẽ bằng hình chữ nhật và đánh nhãn bằng một cặp gồm kiểu khái niệm (Concept Type) và tham chiếu khái niệm (Concept Referent), biểu diễn một thực thể có kiểu và tham chiếu như xác định trong nhãn. Mỗi nút quan hệ, được vẽ bằng hình bầu dục và đánh nhãn bằng một kiểu quan hệ (Relation Type), biểu diễn quan hệ giữa các thực thể định nghĩa bởi các nút khái niệm nối vào nút quan hệ này. Ví dụ, đồ thị khái niệm ở Hình 2 biểu diễn "Nam học một môn về Khoa học Máy tính".




*Hình 2.* Một đồ thị khái niệm ví dụ

Trong ví dụ này, [Sinhviên: Nam], [Môn: *], [Lĩnh_vực: Khoa học Máy tính] là các khái niệm, với Sinh_viên, Môn và Lĩnh_vực là các kiểu khái niệm, trong khi (học) và (về) là các quan hệ, với học và về là các kiểu quan hệ. Các kiểu khái niệm và kiểu quan hệ được định nghĩa trong một Ontology đang xét. Các tham chiếu Nam và Khoa học Máy tính của các khái niệm [Sinh_viên: Nam] và [Lĩnh_vực: Khoa học Máy tính] được gọi là các tham chiếu cá thể (Individual Referent). Tham chiếu "*" của khái niệm [Môn: *] được gọi là tham chiếu chung (Generic Referent), chỉ đến một thực thể không xác định.




*Hình 3.* Một đồ thị khái niệm truy vấn

Một đồ thị khái niệm truy vấn được định nghĩa là một đồ thị khái niệm mà các tham chiếu của nó là một tham chiếu cá thể, tham chiếu chung ký hiệu bằng "*", hoặc tham chiếu truy vấn ký hiệu bằng "?". Tham chiếu chung có thể so trùng với bất kỳ tham chiếu cá thể

nào. Tham chiếu truy vấn biểu diễn tham chiếu đến thực thể muốn tìm kiếm. Mỗi đồ thị khái niệm truy vấn kèm theo các ràng buộc về giá trị của các thuộc tính của các khái niệm trong nó. Ví dụ, Hình 3 mô tả câu truy vấn tìm "Các trường đại học ở một thành phố cấp một ở Việt Nam" trong phần mềm VN-KIM QER.

Đồ thị khái niệm linh hoạt hơn các mẫu truy vấn cố định và dễ đọc-viết hơn SeRQL, nhưng vẫn không thân thiện bằng câu truy vấn viết bằng tiếng Việt như trên. Bổ sung truy vấn bằng ngôn ngữ tự nhiên vào VN-KIM QER là mục tiêu của nghiên cứu trong bài báo này. Các phần tiếp theo đây trình bày cách tiếp cận và phương pháp chuyển câu truy vấn tiếng Việt sang đồ thị khái niệm.

## 3. NHẬN DIỆN THỰC THỂ VÀ TỪ QUAN HỆ

Quá trình biến đổi câu truy vấn thành đồ thị khái niệm thực chất là quá trình xác định các thực thể trong câu truy vấn và các mối quan hệ giữa chúng. Vì vậy, vấn đề có thể được giải quyết theo ba bước sau đây:
*Bước 1*: Nhận diện các thực thể và từ quan hệ (những từ biểu diễn quan hệ giữa các thực thể, như "ở", "tại", "có", "của",...) có trong câu truy vấn.
*Bước 2*: Từ các thực thể và từ quan hệ đã nhận diện được, xây dựng khung cho đồ thị khái niệm, gồm các khái niệm và các quan hệ biểu diễn bằng các từ quan hệ giữa chúng.
*Bước 3*: Từ khung đã xây dựng được, xác định chính xác các kiểu quan hệ giữa các khái niệm, tạo nên một đồ thị khái niệm hoàn chỉnh.

Phần này trình bày Bước 1 và Bước 2, còn Bước 3 được trình bày ở Phần 4 tiếp theo. Có hai loại thực thể cần nhận diện trong một câu truy vấn, là thực thể có tên và thực thể không tên. Ví dụ, trong câu truy vấn tìm "thủ đô của Việt Nam", "Việt Nam" là một thực thể có tên, còn "thủ đô" là một thực thể không tên. Vì vậy, trước hết chúng tôi sử dụng VN-KIM để nhận diện các thực thể có tên. Mỗi thực thể có tên khi được nhận diện đầy đủ sẽ tương ứng với một khái niệm trong đồ thị, trong đó kiểu khái niệm là lớp và tham chiếu khái niệm là tên hoặc định danh của thực thể đó. Ví dụ, thực thể "Việt Nam" sẽ tương ứng với khái niệm [Quốc_gia: Việt Nam].

Trong khi đó việc nhận diện các thực thể không tên có thể được thực hiện dựa trên các tiền tố tên, như "ông" hay "bác sĩ" cho biết thực thể là con người, còn "thành phố" hay "tỉnh" cho biết đó là nơi chốn. Hiện tại VN-KIM có một kho tiền tố tên phân theo từng lớp trong Ontology và được tổ chức thành một Gazetteer trong GATE, một phần mềm mã nguồn mở bao gồm nhiều phần mềm công cụ để xử lý ngôn ngữ tự nhiên ([5]). Chúng tôi sử dụng GATE trên Gazetteer của VN-KIM như một bảng ánh xạ để xác định các thực thể không tên trong câu truy vấn. Mỗi thực thể không tên khi được xác định sẽ tương ứng với một khái niệm có kiểu khái niệm là lớp của thực thể, còn tham chiếu khái niệm là "?" hoặc "*" tùy theo đó là thực thể cần tìm hay là thực thể tuỳ ý. Ví dụ, thực thể "thủ đô" trong truy vấn trên sẽ tương ứng với khái niệm [Thủ_đô: ?].

Trong một câu truy vấn, ngoài các từ biểu diễn các thực thể, còn có các mạo từ ("các", "những", "một", ), liên từ và các từ biểu diễn quan hệ giữa các thực thể. Việc phân biệt giữa các loại từ này cũng có thể được thực hiện thông qua Gazetteer. Sau khi được nhận diện, các mạo từ bị loại bỏ đi, do chúng không mang nhiều ý nghĩa. Vấn đề còn lại là mỗi từ quan hệ xuất hiện biểu diễn quan hệ giữa các thực thể nào trong câu truy vấn. Trên lý

thuyết, việc phân tích các mối quan hệ này có thể dựa trên một văn phạm và một bộ phân tích cú pháp cho một câu ngôn ngữ tự nhiên tổng quát. Tuy nhiên, lời giải tổng quát này sẽ đòi hỏi nhiều thời gian xử lý và cũng không bảo đảm hoàn toàn chính xác do sự nhập nhằng về cả cú pháp và ngữ nghĩa trong ngôn ngữ tự nhiên. Vì vậy, cách tiếp cận của chúng tôi là chỉ xây dựng một văn phạm và bộ phân tích cú pháp bề mặt, đủ hiệu quả cho các cụm từ truy vấn tiếng Việt, có cấu trúc đơn giản hơn nhiều so với các câu tổng quát.

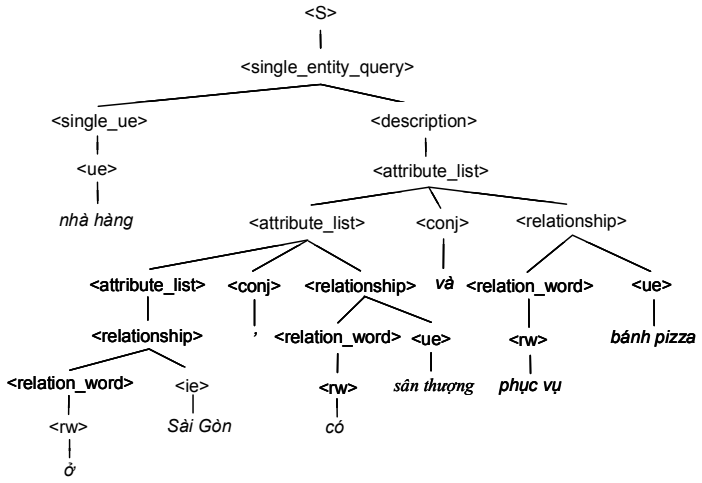
Trong việc xây dựng văn phạm nói trên, cần lưu ý một từ quan hệ không nhất thiết biểu diễn quan hệ giữa hai thực thể cách nhau bởi từ đó. Ví dụ, trong câu truy vấn tìm "thủ đô của một quốc gia ở châu Á", từ "ở" xác định một quan hệ giữa hai thực thể "quốc gia" và "châu Á". Tuy nhiên, trong câu truy vấn tìm "thành phố ở Việt Nam có sân bay", từ "có" lại biểu diễn một quan hệ giữa hai thực thể "thành phố" và "sân bay". Sự khác biệt ở đây là, trong ví dụ thứ nhất, "quốc gia" là một thực thể không tên nên quan hệ với thực thể "châu Á" bổ nghĩa thêm cho nó. Còn trong ví dụ thứ hai, thực thể "Việt Nam" đã được xác định rõ (tương ứng duy nhất một định danh trong cơ sở tri thức), nên tính chất "có sân bay" là để bổ nghĩa cho thực thể không tên "thành phố". Chúng tôi áp dụng Heuristic này trong việc xác định các thực thể tham gia vào một quan hệ. Một vấn đề nữa khi xây dựng văn phạm đó là việc xử lý các cấu trúc song song khi gặp liên từ "và" hoặc dấu phẩy. Những câu truy vấn loại này thường có dạng một chủ thể quan hệ với nhiều khách thể, ví dụ như câu truy vấn tìm "trường đại học ở Việt Nam và có tên là Bách Khoa", hoặc nhiều chủ thể quan hệ với một khách thể, ví dụ như câu truy vấn tìm "Thành phố và trường đại học ở Việt Nam".

```
<S> ::= <single_entity_query> | //truy vấn một thực thể
        <multiple_entity_query> //truy vấn nhiều thực thể
<single_entity_query> ::= <single_ue> <description>
<single_ue> ::= <ue> <relation_word> <ie> | <ue>
<description> ::= <attribute_list> | //danh sách các mối quan hệ được nối với nhau //bởi liên từ
          <relation_word> <single_entity_query> //quan hệ với một thực thể//được mô tả khác
<attribute_list> ::= <attribute_list> <conj> <relationship> | <relationship>
<relationship>::=  <relation_word> <ue> | <relation_word> <ie>
<relation_word> ::= <rw> | ε
<multiple_entity_query> ::= <ue_list> <description>
<ue_list> ::= <ue_list> <conj> <ue> | <ue> <conj> <ue>
```

*Hình 4.* Văn phạm cho các câu truy vấn thực thể

Dựa trên những ý tưởng trên, chúng tôi đã xây dựng một văn phạm cho các câu truy vấn như sau. Gọi <ie> là thực thể xác định, <ue> là thực thể không xác định, <rw> là từ quan hệ, <conj> là liên từ và <S> là ký hiệu khởi đầu. Văn phạm cho câu truy vấn được xây dựng như trong Hình 4, trong đó phần nằm sau ký hiệu // là chú thích và ký hiệu $\varepsilon$ biểu diễn chuỗi rỗng. Luật sinh cho <S> có hai dạng: (1) truy vấn một thực thể, ví dụ như tìm "thủ đô của một quốc gia ở châu Á", trong đó "thủ đô" là thực thể cần tìm, được đánh dấu bằng tham chiếu "?" trong đồ thị khái niệm; hoặc (2) truy vấn nhiều thực thể, ví dụ như tìm "cha và mẹ của ông Nguyễn Văn A", trong đó các thực thể cần tìm là "cha" và "mẹ". Trong luật sinh cho <single_entity_query>, biến <single_ue> biểu diễn một thực thể không xác định có thể được bổ nghĩa bởi một thực thể xác định khác. Biến <description> biểu diễn phần đặc tả cho thực thể cần truy vấn. Luật sinh cho <multiple_entity_query> cũng tương tự, chỉ khác là có nhiều hơn một thực thể cần truy vấn được biểu diễn bằng biến <ue_list>. Hình

5 cho thấy cây dẫn xuất của câu truy vấn "nhà hàng ở Sài Gòn, có sân thượng và phục vụ bánh pizza".




*Hình 5.* Cây dẫn xuất của một câu truy vấn thực thể

Như vậy, sau Bước 2, kết quả thu được là một khung đồ thị khái niệm bao gồm các thông tin sau:

1. Các khái niệm có kiểu thuộc Ontology của VN-KIM.

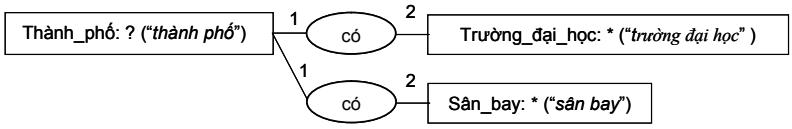
2. Một số thực thể đã được định danh, là những thực thể hiện có trong cơ sở tri thức của VN-KIM

3. Các thực thể chưa định danh và cần truy vấn có tham chiếu khái niệm là "?"; các thực thể chưa định danh còn lại có tham chiếu là "*"

4. Tất cả các quan hệ đều mới ở dạng từ quan hệ, chưa được ánh xạ vào một kiểu quan hệ nào trong Ontology của VN-KIM.

5. Chuỗi kí tự tương ứng với mỗi thực thể trong câu truy vấn được lưu lại cùng với khái niệm biểu diễn thực thể đó, để phân giải sự nhập nhằng nếu có ởbước tiếp theo trong quá trình sinh đồthị khái niệm hoàn chỉnh.

Hình 6 minh họa khung đồ thị khái niệm cho câu truy vấn tìm " các thành phố có trường đại học và có sân bay" , trong đó thành phần trong dấu ngoặc đơn là chuỗi kí tự ban đầu tương ứng với một thực thể.




*Hình 6.* Khung đồ thị khái niệm sinh ra từ một câu truy vấn tiếng Việt

## 4. XÁC ĐỊNH KIỂU QUAN HỆ GIỮA CÁC KHÁI NIỆM

Một cách tổng quát, việc chuyển một từ quan hệ sang một kiểu quan hệ có thể được thực hiện thông qua một bảng ánh xạ. Tuy nhiên, trên thực tế một từ quan hệ có thể biểu diễn nhiều kiểu quan hệ và một kiểu quan hệ cũng có thể được diễn đạt bằng nhiều từ quan hệ khác nhau. Do đó, để xác định đúng kiểu quan hệ tương ứng với một từ quan hệ, chúng tôi

khai thác thông tin về các thực thể đã thu thập được từ hai bước trước để giới hạn các kiểu quan hệ có thể. Thông tin đó gồm lớp (hay kiểu) của các thực thể tham gia vào mối quan hệ và các chuỗi kí tự biểu diễn thực thể trong câu truy vấn.

Xét những mối quan hệ có dạng:

$$[C_1 \times S_1] - w - [C_2 \times S_2]$$

trong đó $w$ là từ quan hệ giữa hai thực thể, $C_1$ và $C_2$ lần lượt là lớp của thực thể thứ nhất và thứ hai, và $S_1$ và $S_2$ lần lượt là chuỗi kí tự biểu diễn thực thể thứ nhất và thứ hai. Chúng tôi lập các ánh xạ sau đây:

+ Ánh xạ $w$ sang $R_1$, là tập những quan hệ có thể có tương ứng với $w$. Ví dụ từ quan hệ "ở" tương ứng với tập {được_định_vị_ở, là_vùng_con_của,...}.

+ Ánh xạ $(C_1, C_2)$ sang $R_2$, là tập những quan hệ có thể có giữa hai lớp thực thể $C_1$ và $C_2$. Ví dụ (Con_người, Con_người) có thể có các quan hệ thuộc {có_cha, có_mẹ, có_người_thân,...}.

+ Ánh xạ $(S_1, w)$ sang $R_3$, là tập những quan hệ có thể có nếu biết từ quan hệ $w$ và chuỗi kí tự tương ứng với chủ thể của quan hệ. Ví dụ ("cha", "của") rút ra từ câu truy vấn tìm "cha của ông Nguyễn Văn A" có thể xác định được quan hệ là có_cha.

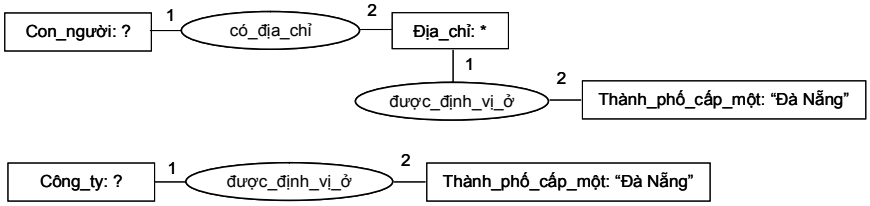
+ Ánh xạ $(w, S_2)$ sang $R_4$, là tập những quan hệ có thể có nếu biết từ quan hệ $w$ và chuỗi kí tự tương ứng với khách thể của quan hệ. Ví dụ ("có", "mẹ") rút ra từ câu truy vấn tìm "ca sĩ có mẹ là bà Nguyễn Thị B"có thể xác định được quan hệ là có_mẹ.

Khi đó, kiểu quan hệ thật sự tương ứng với từ quan hệ w thuộc vào phần giao của $R_1, R_2, R_3$, và $R_4$. Nếu phần giao đó chỉ còn một kiểu quan hệ duy nhất thì đó là lời giải. Trong trường hợp vẫn còn nhập nhằng, chúng tôi khai thác thêm các quan hệ hoặc thực thể khác đứng gần để phân giải. Ví dụ với câu truy vấn tìm "người có con là chồng bà Nguyễn Thị B", từ quan hệ "có con" có thể tương ứng với tập các quan hệ {có_con_trai, có_con_gái} trong Ontology đang xét. Tuy nhiên, do có quan hệ "là chồng" đứng gần, kiểu quan hệ có_con_gái bị loại trừ.

Các luật phân giải này được biểu diễn như các luật sinh có dạng:

**Nếu [điều kiện] thì [hành động]**

Thành phần [điều kiện] đặc tả các ràng buộc về: (1) các từ quan hệ và kiểu quan hệ; và (2) các lớp, định danh, hoặc chuỗi kí tự biểu diễn thực thể. Thành phần [hành động] cho phép: (1) thay đổi các lớp hoặc định danh của các thực thể; (2) thay đổi các kiểu quan hệ; và (3) thêm hoặc xóa các thực thể hoặc quan hệ trong đồ thị khái niệm đang được sinh ra. Mỗi luật được gán một độ ưu tiên, với giá trị mặc định là 0. Khi xử lý, các luật sẽ lần lượt được áp dụng theo độ ưu tiên từ cao xuống thấp.




*Hình 7.* Các đồ thị khái niệm kết quả khác nhau do sự đa nghĩa của từ quan hệ "ở"

Ví dụ cho thấy một trường hợp nhập nhằng được phân giải bằng luật sinh là câu truy vấn tìm "người ở Đà Nẵng". So sánh câu này với câu truy vấn tìm "công ty ở Đà Nẵng",

mặc dù hai câu có cấu trúc tương tự nhau, nhưng hai đồ thị khái niệm truy vấn tương ứng lại khác nhau như minh họa ở Hình 7. Lý do là từ quan hệ "ở" trong truy vấn thứ hai tương ứng với quan hệ được_định_vị_ở, chỉ áp dụng cho tổ chức hoặc nơi chốn, trong khi từ quan hệ "ở" trong truy vấn thứ nhất có nghĩa là địa chỉ. Tức là hai câu truy vấn cần được dịch sang đồ thị khái niệm theo hai cách khác nhau tuỳ theo lớp của thực thể cần tìm.

```
<premise>
      <subject  var  =  "xxx"  className  =  "&vnkimo;Con_người"/>
      <relation var = "yyy" value = "ở,tại"/>
      <object var = "zzz" className = "&vnkimo;Nơi_chốn"/>
</premise>
<consequent>
      <subject var = "xxx" />
      <relation uri = "&vnkimo;có_địa_chỉ" value = "có_địa_chỉ"/>
      <object var = "ttt" className = "&vnkimo;Địa_chỉ"
                  displayValue = "*"/>
</consequent>
<consequent>
      <subject var = "ttt" />
      <relation uri = "&vnkimo;được_định_vị_ở"
                     value = "được_định_vị_ở"/>
      <object var = "zzz"/>
</consequent>
<delete var = "yyy"/>
</rule>
```

*Hình 8.* Luật sinh xử lý một trường hợp nhập nhằng ngữ nghĩa của từ quan hệ

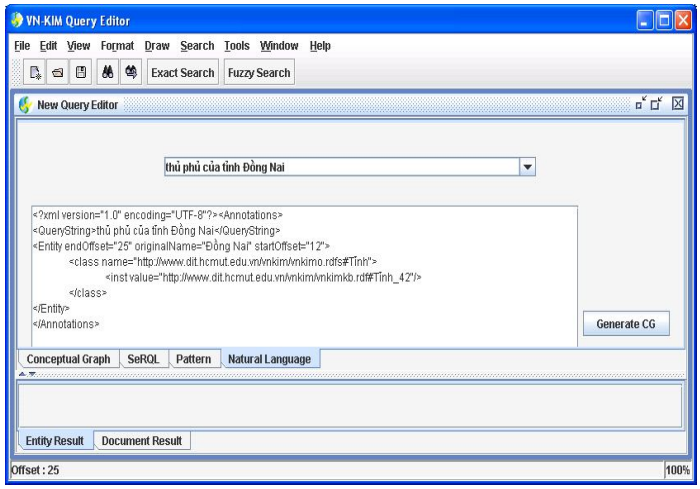
Hình 8 trình bày luật sinh (định dạng XML) xử lý truy vấn thuộc trường hợp thứ nhất. Ở đây, thành phần điều kiện được đóng trong Tag <premise>, còn thành phần hành động được đóng trong Tag <consequent>. Ý nghĩa của luật này là, nếu trong đồ thị khái niệm có một quan hệ mà chủ thể (trong Tag <subject>) thuộc lớp Con_người, từ quan hệ (trong Tag <relation>) có giá trị là "ở" hoặc "tại", và đối tượng (trong Tag <object>) thuộc lớp Nơi_chốn, thì thực hiện những hành động sau: (1) tạo một thực thể mới (ứng với biến "ttt") thuộc lớp Địa_chỉ và có tham chiếu là "*" (thực thể tuỳ ý); (2) gắn thực thể này với chủ thể của quan hệ ban đầu thông qua quan hệ có_địa_chỉ, và gắn với khách thể của quan hệ ban đầu thông qua quan hệ được_định_vị_ở; (3) xóa quan hệ giữa hai thực thể ban đầu đi. Kết quả ta sẽ thu được đồ thị khái niệm như ở Hình 7.

## 5. HIỆN THỰC VÀ ĐÁNH GIÁ KẾT QUẢ

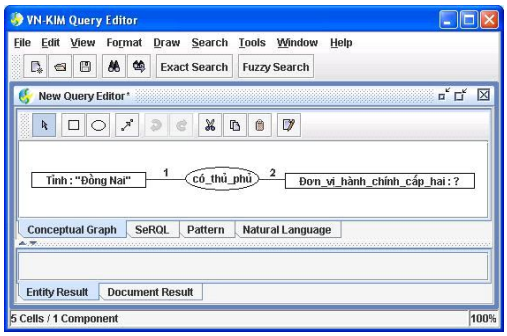
Hệ thống luật sinh đồ thị khái niệm trình bày ở trên đã được tích hợp vào phần mềm truy hồi thông tin VN-KIM QER, cho phép người sử dụng truy vấn bằng các cụm từ tiếng Việt thông thường. Hình 9 cho thấy giao diện của phần mềm, trong đó hộp văn bản đang hiển thị truy vấn tìm "thủ phủ của tỉnh Đồng Nai". Hộp văn bản bên dưới hiển thị kết quả nhận diện thực thể có tên của VN-KIM, cho thấy "Đồng Nai" đã được nhận diện là một thực thể có trong cơ sở tri thức (định danh #Tỉnh_42). Nút Generate CG dùng để yêu cầu thực hiện việc dịch câu truy vấn đó sang đồ thị khái niệm, và kết quả được hiển thị như trong Hình 10.

Phần mềm đã được thử nghiệm trên 65 câu truy vấn tiếng Việt có dạng thông dụng. Trong đó có 51 câu được VN-KIM nhận diện đầy đủ và chính xác tất cả các thực thể có tên xuất hiện, và phần mềm dịch đúng được 48 câu, như liệt kê ở Bảng 1. Còn trong số 14 câu mà VN-KIM nhận diện thiếu và sai một số thực thể có tên, phần mềm có thể sửa chữa và

dịch đúng được 3 câu. Như vậy, nếu đầu vào đã có đúng và đủ tất cả các thực thể có tên, thì độ chính xác của phần biên dịch sang đồ thị khái niệm là 48/51 =∼94%.




*Hình 9.* Giao diện truy vấn bằng tiếng Việt của VN-KIM QER




*Hình 10.* Giao diện hiển thị đồ thị khái niệm kết quả của VN-KIM QER

## 6. KẾT LUẬN

Chúng tôi đã trình bày một giải pháp biến đổi truy vấn từ ngôn ngữ tự nhiên, tiếng Việt, sang đồ thị khái niệm để truy hồi thông tin. Quá trình được chia thành ba giai đoạn là nhận diện thực thể và từ quan hệ, xây dựng khung đồ thị, và hoàn chỉnh đồ thị khái niệm kết quả. Bước một sử dụng lại VN-KIM để nhận diện các thực thể có tên và phần mềm GATE để nhận diện các thực thể không tên và từ quan hệ trong câu truy vấn. Bước hai áp dụng một văn phạm cho những dạng câu truy vấn tiếng Việt thông dụng và một bộ phân tích cú pháp để xác định các thực thể liên kết với nhau qua các từ quan hệ. Bước ba sử dụng các bảng ánh xạ quan hệ kết hợp với các luật sinh phân giải nhập nhằng để xác định chính xác các kiểu quan hệ trong Ontology tương ứng với các từ quan hệ trong câu truy vấn, hoàn chỉnh đồ thị khái niệm kết quả.

Giải pháp đã được tích hợp vào phần mềm VN-KIM QER và thí nghiệm cho thấy kết quả dịch tốt nếu đầu vào, sau khi nhận diện các thực thể có tên, ít bị nhiễu sai sót. Tuy vậy, hệ thống luật sinh cần được tiếp tục bổ sung để xử lý thêm các dạng câu truy vấn tiếng Việt

khác. Bên cạnh đó chúng tôi cũng đang nghiên cứu các phương pháp học máy thống kê để bổ trợ cho hệ thống luật trong việc xác định chính xác các quan hệ giữa các thực thể trong truy vấn.

*Bảng 1.* Các câu truy vấn được VN-KIM nhận diện và dịch đúng sang đồ thị khái niệm

| | |
|---|---|
| 1. Các huyện ở thành phố cấp một ở nước Việt Nam | 25. Nơi sinh của Bác Hồ |
| 2. Các trường đại học ở thành phố Hồ Chí Minh và Hà Nội | 26. Nơi sinh của chủ tịch Hồ Chí Minh |
| 3. Các trường đại học ở Việt Nam | 27. Nơi sinh của ông Lê Quí Đôn quê ở tỉnh Thái Bình |
| 4. Cha của Hồ Chí Minh | 28. Ông Lê Quí Đôn quê ở tỉnh Thái Bình |
| 5. Cha mẹ của vua Lê Thánh Tôn | 29. Ông Nguyễn Văn A ở Đà Nẵng |
| 6. Cha và mẹ của ông A ở Sài Gòn | 30. Ông Trần Quốc Tuấn |
| 7. Con của ông Nguyễn Tấn Dũng | 31. Quận Tân Bình thành phố Hồ Chí Minh |
| 8. Con trai của ông Nguyễn Tấn Dũng | 32. Quốc khánh nước Mỹ |
| 9. Các công ty ở đường Trần Hưng Đạo ở Hà Nội | 33. Sản phẩm của công ty Bình Minh |
| 10. Các công ty ở phường 13 quận 9 Hà Nội | 34. Các thành phố có trường đại học và có sân bay |
| 11. Các dịch vụ chăm sóc sắc đẹp ở một thành phố của VN | 35. Các thành phố ở Việt Nam có sân bay |
| 12. Đại học Quốc Gia thành phố Hồ Chí Minh | 36. Các thành viên của Quốc hội nước Việt Nam |
| 13. Đảng Cộng sản Việt Nam | 37. Thủ đô của một quốc gia ở châu Á |
| 14. Địa chỉ của trường đại học Bách Khoa thành phố HCM | 38. Thủ đô của Việt Nam |
| 15. Địa chỉ trang web công ty Kyoshin Việt Nam | 39. Thủ phủ của Đồng Nai |
| 16. Đơn vị tiền tệ của Thái Lan | 40. Thủ phủ của tỉnh Đồng Nai |
| 17. Đường Trần Hưng Đạo | 41. Tỉnh của Việt Nam có thủ phủ là thị xã Trà Vinh |
| 18. Đường Trần Hưng Đạo ở Hà Nội | 42. Tỉnh của Việt Nam có thủ phủ là Trà Vinh |
| 19. Các khách sạn ở đường Trần Hưng Đạo ở Hà Nội | 43. Tổng bí thư của Đảng Cộng sản Việt Nam |
| 20. Lãnh đạo của Hà Nội | 44. Trang Web của báo Người Lao động |
| 21. Giám đốc công ty Ánh Sáng | 45. Trang Web của công ty điện tử viễn thông Sony |
| 22. Người có con là Nguyễn Văn An | 46. Trang Web của các công ty điện tử viễn thông |
| 23. Các khu công nghiệp ở Việt Nam | 47. Đền Trần Hưng Đạo |
| 24. Các thành phố cấp một ở Việt Nam | 48. Trường đại học Bách Khoa thành phố Hồ Chí Minh |

## TÀI LIỆU THAM KHẢO


[1] N. K. Anh, Translating logical queries into SQL queries in natural language query systems, *Proceedings of the 3rd National Symposium on Research, Development and Application of Information and Communication Technology (ICT.rda'06)*, Science and Technics Publishing House, Ha Noi, Vietnam, May 5-6, 2006 (124–130).

[2] T. Berners-Lee, J. Hendler, O. Lassila, The semantic web, *Scientific American Journal* **284** (5) (2001) 34–43.

[3] T. H. Cao, Xây dựng và phát triển các kỹ thuật khai thác thông tin trên Web có ngữ nghĩa, "Đề tài KC01.21" (2004-2006).

[4] T. H. Cao, H. T. Do, B. T. N. Pham, and D. Q. Vu, Conceptual graphs for knowledge querying in VN-KIM, *Contributions of the 13th International Conference on Conceptual Structures* Kassel, Germany, July 18-22, 2005 (27–40).

[5] H. Cunningham, et. al, GATE: A Framework and graphical development environment for robust NLP tools and applications, *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)*, Philadelphia, USA, July 7-12, 2002.

[6] R. Guha, R. McCool, and E. Miller, Semantic search, *Proceedings of the 12th International Conference on World Wide Web*, Budapest, Hungary, May 20-24, 2003 (700–709).

[7] S. Handschuh, S. Staab, and F. Ciravegna, S-CREAM: semi-automatic CREAtion of metadata, *Proceedings of the 13th International Conference on Knowledge Engineering and Management*, Siguenza, Spain, October 1-4, 2002.

[8] S. Hensman, and J. Dunnion, Using linguistic resources to construct conceptual graph representation of texts, *Proceedings of the 7th International Conference on Text, Speech and Dialogue, LNAI* **3206** Springer-Verlag, Brno, Czech Republic, September 8-11, 2004 (81–88).

[9] A. Kampman, F. Harmelen, and J. Broekstra, Sesame: a generic architecture for storing and querying RDF and RDF schema, *Proceedings of the 1st International Semantic Web Conference, LNCS* **2342** Springer-Verlag, Sardinia, Italia, June 9-12, 2002 (54–68).

[10] A. Kiryakov, B. Popov, I. Terziev, D. Manov, and D. Ognyanoff, Semantic annotation, indexing, and retrieval, *Journal of Web Semantics* **2** (1) (2005).

[11] T-V. T. Nguyen, and T. H. Cao, VN-KIM IE: Automatic extraction of Vietnamese entities on the Web, *New Generation Computing Journal* **25** (3) (2007) 277–292.

[12] N. F. Noy, M. Sintek, S. Decker, M. Crubezy, R. W. Fergerson, and M. A. Musen, Creating semantic web contents with Protégé-2000, *IEEE Intelligent Systems* **2** (16) (2001) 60–71.

[13] S. Shady, F. Karray, M. Kamel, Enhancing text retrieval performance using conceptual ontological graph, *Proceedings of the 6th IEEE International Conference on Data Mining*, Hong Kong, China, December 18-22, 2006 (39–44).

[14] J. F. Sowa, *Conceptual strutures: Information Processing in Mind and Machine*, Addison-Wesley, 1984.

[15] J. F. Sowa, *Matching logical structure to linguistic structure*, N. Houser, D. D. Roberts, and Van J. Evra, *Studies in the Logic of Charles Sanders Peirce*, (Eds) Indiana University Press, 1997 (418–444 ).

[16] M. Vargas-Vera, et. al., MnM: ontology driven semi-automatic support for semantic markup, *Proceedings of the 13th International Conference on Knowledge Engineering and Management*, Siguenza, Spain, October 1- 4, 2002.

[17] H. Yao, and L. Etzkorn, Conversion from the conceptual graph (CG) model to the resource description framework (RDF) model, *Contributions of the 12th International Conference on Conceptual Structures*, Huntsville, AL, USA, July 19-23, 2004 (98–114).

[18] L. Zhang, and Y. Yu, Learning to generate CGs for domain specific sentences, *Proceedings of the 9th International Conference on Conceptual Structures, LNAI* **2120** Springer-Verlag, Standford, CA, USA, July 30 - August 3, 2001 (44–57).

*Nhận bài ngày 16 - 1 - 2007*
*Nhận lại sau sửa ngày 6 - 9 -2007*