

TRUY VẤN NGÔN NGỮ TỰ NHIÊN KHÔNG HOÀN CHỈNH ĐỐI VỚI CÁC CƠ SỞ DỮ LIỆU QUAN HỆ

IMPERFECT NATURAL LANGUAGE QUERIES TO RELATIONAL DATABASES

Nguyễn Kim Anh, Phạm Thị Thu Hoài
Khoa Công nghệ thông tin, ĐHBK Hà Nội

TÓM TẮT

Ngôn ngữ tự nhiên là một trong các phương tiện truyền thông mạnh nhất. Một lĩnh vực mà hệ xử lý ngôn ngữ tự nhiên (NLP) tỏ ra có hiệu quả là hệ truy vấn cơ sở dữ liệu. Bài báo này trình bày một cách tiếp cận biến đổi các truy vấn ngôn ngữ tự nhiên không hoàn chỉnh đối với các cơ sở dữ liệu quan hệ thành SQL và sau đó cơ sở dữ liệu này có thể được tra cứu và một câu trả lời thích hợp có thể được sinh ra.

ABSTRACT

Natural language is one of the most powerful mediums for communication. One area in which Natural Language Processing (NLP) system are powerful enough to be effective is database query system. This paper presents an approach to convert imperfect natural language queries to relational databases into SQL and then the database can be consulted and an appropriate response can be generated.

1. GIỚI THIỆU

Ngày nay máy tính đã trở thành một công cụ phổ biến và hỗ trợ đắc lực cho các hoạt động của con người. Tuy vậy, đối với người dùng không chuyên, giao tiếp với máy tính là một công việc khó khăn và đòi hỏi phải qua một quá trình đào tạo tốn kém bởi vì họ cần nắm được một dạng ngôn ngữ trung gian mà máy tính có thể “hiểu” được. Bởi vậy, một yêu cầu đặt ra từ rất sớm cho khoa học máy tính là làm thế nào giúp con người có thể giao tiếp với máy tính càng đơn giản càng tốt và có lẽ không gì thuận lợi hơn nếu con người có thể giao tiếp với máy tính bằng chính ngôn ngữ của mình. Để thực hiện được điều đó đòi hỏi phải có các hệ thống trung gian giúp máy tính có khả năng “hiểu” được ngôn ngữ của con người. Các hệ thống xử lý ngôn ngữ tự nhiên (Natural Language Processing - NLP) được ra đời nhằm để đáp ứng yêu cầu này.

Một lĩnh vực có thể áp dụng hiệu quả những kết quả nghiên cứu của NLP là các hệ truy vấn cơ sở dữ liệu [5]. Lý do là các cơ sở dữ liệu thường phủ một miền tri thức đủ nhỏ nên trong nhiều tình huống, sự nhập nhằng trong ngôn ngữ tự nhiên nói trên có

thể giải quyết được. Các hệ thống xử lý ngôn ngữ tự nhiên dùng cho truy vấn cơ sở dữ liệu được gọi chung là *Giao diện ngôn ngữ tự nhiên với cơ sở dữ liệu* (Natural Language Interface to Database - NLIDB)[2]. NLIDB phải trợ giúp tối đa cho người dùng trong khả năng của nó, tức là:

- Phải có khả năng tự giải quyết những lỗi không quan trọng mà cách khắc phục là rõ ràng.
- Phải nhận biết tương đối đầy đủ các mẫu câu có thể đưa vào.
- Phải đưa ra các phản hồi càng rõ ràng chính xác càng tốt để người dùng có thể yên tâm về cách giải quyết của hệ thống.

Bài báo này trình bày một cách tiếp cận xây dựng hệ thống cho phép truy vấn bằng ngôn ngữ tự nhiên tiếng Việt có khả năng xử lý các truy vấn tự nhiên không hoàn chỉnh. Nội dung bài báo được trình bày như sau: phần 2 mở đầu với một số khái niệm cơ bản liên quan đến việc phân tích cú pháp và ngữ nghĩa các câu truy vấn

tự nhiên. Phần 3 trình bày một kiến trúc phân tích của hệ thống truy vấn ngôn ngữ tự nhiên làm cơ sở cho các kỹ thuật xử lý các câu truy vấn tự nhiên không hoàn chỉnh. Phần 4 trình bày các kiểu truy vấn không hoàn chỉnh và kỹ thuật xử lý. Cuối cùng, phần 5 trình bày một ví dụ minh họa và phần 6 đưa ra một vài đánh giá và kết luận.

2. MỘT SỐ KHÁI NIỆM CƠ BẢN

2.1. Sơ đồ thực thể-liên kết

Trong thực tế, khi thiết kế cơ sở dữ liệu (CSDL) quan hệ cho một xí nghiệp, chúng ta thường sử dụng một sơ đồ thực thể-liên kết biểu diễn cấu trúc logic tổng thể của CSDL đối với xí nghiệp này. Các thành phần cơ bản của một sơ đồ thực thể-liên kết là các thực thể, các thuộc tính và các liên kết. Một tập thực thể (gọi đơn giản là thực thể) được gán một tên gọi là một danh từ. Mỗi một thuộc tính của thực thể được gán một tên gọi cũng là một danh từ. Mỗi liên kết được gán một tên gọi là một động từ. Thông thường, ngữ nghĩa của các thực thể, các thuộc tính và các liên kết đã phần nào được phản ánh thông qua tên gọi của chúng [3]. Do vậy, sơ đồ thực thể-liên kết đối với một xí nghiệp có một ý nghĩa quan trọng nhất định đối với bộ phân tích cú pháp cũng như bộ phân tích và xử lý ngữ nghĩa để hiểu nghĩa và xử lý ngữ nghĩa không hoàn chỉnh của các câu truy vấn đối với CSDL của xí nghiệp này và đối với chúng tôi, sơ đồ thực thể-liên kết đối với một xí nghiệp có thể được xem như là những tri thức về ngữ nghĩa đã được biết về CSDL mà chúng ta đang xem xét. Đồng thời, sơ đồ thực thể-liên kết này cũng được sử dụng để ánh xạ vào mô hình dữ liệu quan hệ đối với xí nghiệp khi thiết kế cơ sở dữ liệu (CSDL) quan hệ cho xí nghiệp này. Để đơn giản, chúng tôi giả thiết tên của các quan hệ và tập thuộc tính của chúng được đặt trùng với các tên gọi tương ứng trong sơ đồ thực thể-liên kết được sử dụng khi thực hiện ánh xạ.

2.2. Văn phạm ngữ nghĩa (Semantic Grammar -SG)

Về mặt cú pháp, mỗi câu truy vấn của người sử dụng đều có một dạng văn phạm, mỗi vị trí của một từ hay cụm từ trong câu truy vấn có một phạm trù nhất định, phạm trù đó có thể là đối tượng mà người sử dụng cần hỏi, thuộc tính của đối tượng, giá trị mà người sử dụng đưa vào hay một liên kết giữa các thực thể. Việc xác định tập các luật cho phép sản sinh ra tập các mẫu truy vấn hạn chế là có thể và được gọi là văn phạm ngữ nghĩa. Trong [4], các tác giả đã xây dựng một tập các mẫu truy vấn theo SG cố định và phụ thuộc miền ứng dụng. Ở đây, chúng tôi chỉ xét một ngôn ngữ truy vấn tự nhiên hạn chế với các truy vấn có dạng ‘ [Hãy] đưa ra/tìm/liệt kê/cho biết/.....’ hay có dạng ‘Ai/Cái gì/.....?’ có nghĩa là các truy vấn có dạng mệnh lệnh thức hay có từ để hỏi đặt ở đầu câu. Theo chúng tôi, ngôn ngữ truy vấn tự nhiên hạn chế này là đủ mạnh để bao phủ một lượng lớn các truy vấn thường gặp ở người dùng. Các kỹ thuật biến đổi các truy vấn tự nhiên không hạn chế thành các truy vấn tự nhiên hạn chế không thuộc phạm vi của bài báo này.

Văn phạm ngữ nghĩa được đặc tả một cách hình thức với một tập các cấu trúc kí hiệu:

- **SemanticRole** := Relationship1 [IntermediateRelationship2]- kí hiệu một đường dẫn giữa hai thực thể với một hoặc hai liên kết. Trong trường hợp hai liên kết, ít nhất một liên kết (thường là liên kết đầu tiên) phải được cho trong truy vấn.
- **EntitySelected** := [EntityName] {[EntityAttributeName] [Operator] EntityAttributeValue}- kí hiệu tên thực thể và một danh sách các cặp thuộc tính, phép so sánh và giá trị được sử dụng trong biểu thức chọn. Các tên thuộc tính hoặc thậm chí cả tên thực thể và phép so sánh có thể khuyết trong truy vấn nhưng tất cả các giá trị phải được cho một cách tường minh trong truy vấn.
- **Entity** := EntityName - kí hiệu tên thực thể. Các thực thể có chứa các thuộc

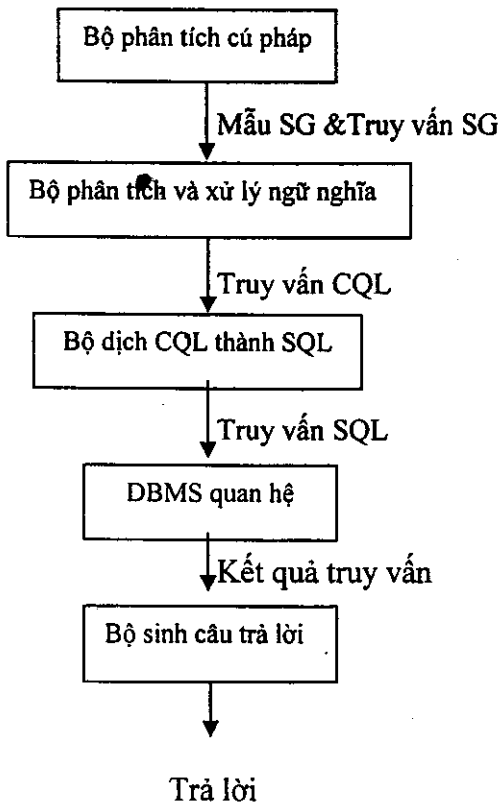
tính cần đưa ra được gọi là các thực thể đích (Target) và các thực thể không chứa các thuộc tính cần đưa ra được gọi là các thực thể nguồn (Source).

Chúng tôi đã xây dựng một bộ luật cho SG để phục vụ cho việc phân tích cú pháp các câu truy vấn tự nhiên hạn chế.

3. KIẾN TRÚC HỆ THỐNG

Trong phần này, chúng tôi sẽ trình bày một kiến trúc phác thảo đối với hệ truy vấn ngôn ngữ tự nhiên và một số phân tích liên quan đến xử lý các câu truy vấn tự nhiên không hoàn chỉnh.

Câu truy vấn ngôn ngữ tự nhiên



Hình 1: Kiến trúc hệ thống

Theo kiến trúc trong hình 1, câu truy vấn ngôn ngữ tự nhiên trước tiên được phân tích bởi bộ phân tích cú pháp. Bộ phân tích cú pháp tham chiếu đến từ điển từ vựng để phân tách các từ có nghĩa trong câu truy vấn tự nhiên, xác định loại từ và từng bước tạo nên mẫu truy vấn theo SG (mẫu SG) và cấu trúc câu truy vấn theo SG (truy vấn SG) đối với câu truy vấn thông qua một tập các

luật SG. Để hạn chế số các mẫu SG có thể đối với câu truy vấn tự nhiên, bộ phân tích cú pháp tra cứu từ điển đồng nghĩa và từ điển phạm trù từ để xác định từ gốc và phạm trù ngữ nghĩa của từ và từ đó cho phép loại trừ các mẫu không phù hợp. Tiếp sau đó, các mẫu SG kết quả được xử lý bởi bộ phân tích và xử lý ngữ nghĩa để hiểu nghĩa của câu truy vấn. Bộ phân tích và xử lý ngữ nghĩa tham chiếu đến sơ đồ thực thể-liên kết và CSDL quan hệ để phân tích các mối quan hệ giữa các thực thể và làm rõ nghĩa các phần tử trong truy vấn SG. Ngôn ngữ được lựa chọn để biểu diễn dạng trung gian của các câu truy vấn là ngôn ngữ truy vấn dựa trên khái niệm (Concept Query Language-CQL[4]). Bộ phân tích ngữ nghĩa biểu diễn ý nghĩa của câu truy vấn đầu vào bằng CQL gồm bốn thành phần cơ bản $Q(tE;sE;\{C(sel),C(sem)\})$ để dễ dàng chuyển đổi thành câu truy vấn SQL và tiến hành truy xuất dữ liệu với:

- tE: chứa các thông tin mà người dùng cần tìm, các thông tin này có thể là các thuộc tính của các thực thể hoặc các hàm kết tập.
- sE: chứa các thông tin về các thực thể mà người dùng cung cấp thông tin (gọi là Source) hoặc các bảng trung gian (Intermediate) mà hệ thống cần phải xác định.
- C(sel) : các điều kiện chọn của người dùng.
- C(sem): các quan hệ ngữ nghĩa giữa các thực thể.

Tiếp theo, câu truy vấn CQL này sẽ được dịch thành một truy vấn SQL mà có thể được thực hiện bởi một phần mềm hệ quản trị CSDL quan hệ nào đó có hỗ trợ SQL. Bộ sinh câu trả lời sử dụng các kết quả của truy vấn SQL để đưa ra câu trả lời cho người sử dụng.

Qua phần trình bày trên, có thể thấy rằng, các câu truy vấn tự nhiên không hoàn chỉnh có thể là các câu truy vấn không xác định được mẫu SG phù hợp (sai về cú

pháp) hay không chuyển được thành một truy vấn CQL (sai về ngữ nghĩa). Trong phần tiếp theo, chúng tôi sẽ đưa ra các kiểu truy vấn tự nhiên không hoàn chỉnh và kỹ thuật khắc phục nó.

4. CÁC KIỂU CÂU TRUY VẤN TỰ NHIÊN KHÔNG HOÀN CHỈNH VÀ KỸ THUẬT XỬ LÝ

Từ góc độ xử lý truy vấn, hai nhân tố quan trọng sau ảnh hưởng đến sự chính xác của truy vấn được phát biểu: các thuật ngữ hay các giá trị được sử dụng để mô tả cái mà người dùng muốn tra cứu và logic gắn kết các giá trị khác nhau trong truy vấn. Ý nghĩa rõ ràng của một truy vấn, do vậy, có thể bị sai lệch bởi một trong hai nhân tố này. Sự điều chỉnh trở thành cần thiết khi tồn tại những lỗ hổng về cú pháp hay ngữ nghĩa giữa truy vấn được phát biểu và ý đồ của người dùng. Mục đích của sự điều chỉnh là mang lại sự phù hợp cả về cú pháp và ngữ nghĩa giữa truy vấn được phát biểu và truy vấn dự định của người dùng. Chúng tôi quan sát thấy sự điều chỉnh truy vấn gồm hai kỹ thuật: sửa đổi và làm mịn.

- (1) Sửa đổi truy vấn được thực hiện đối với các mệnh đề của truy vấn đúng cú pháp hay có ít nhất một mẫu SG phù hợp với truy vấn nhưng chưa rõ về ngữ nghĩa. Các truy vấn này thường chứa các thuật ngữ hay các giá trị mà hệ thống chưa nhận biết được. Kỹ thuật sửa đổi truy vấn bao gồm:
 - Bảo trì mẫu SG của mệnh đề truy vấn mà chỉ làm rõ nghĩa các phần tử chưa biết của mẫu.
 - Tìm một mẫu SG khác với mẫu SG của mệnh đề truy vấn nhưng có liên quan đến nó. Thuật ngữ có liên quan ở đây được hiểu là hai mẫu SG này có một số đoạn chung.
- (2) Làm mịn truy vấn được thực hiện đối với các mệnh đề của truy vấn sai cú pháp hay không đầy đủ và còn được gọi là thiếu thông tin. Các mẫu SG có thể đối với các mệnh đề của truy vấn này thường chứa nhiều đoạn thiếu hay bị đứt đoạn. Quá trình làm mịn theo nghĩa

này thực chất là quá trình bổ sung các thông tin thiếu để lấp kín các lỗ hổng cả về cú pháp và ngữ nghĩa trong các truy vấn đầy đủ bộ phận.

Việc xử lý các truy vấn không hoàn chỉnh phụ thuộc vào bản chất của sự không hoàn chỉnh. Tiếp theo, chúng tôi sẽ đi vào phân tích từng kiểu không hoàn chỉnh và từ đó đưa ra kỹ thuật xử lý chúng.

(1) Các truy vấn thiếu thông tin.

- Thiếu thông tin ở cuối.

Các truy vấn thuộc loại này có mẫu SG thiếu đoạn cuối. Một cách tiếp cận tương tác với người dùng sẽ được áp dụng để bổ sung dần dần các phần tử thiếu. Phần tử thiếu ở đây chỉ có phạm trù là 'thực thể'. Nếu phần tử thiếu có phạm trù là 'vai trò ngữ nghĩa', nó sẽ được xử lý như trường hợp mẫu SG bị phân đoạn. Cách tiếp cận này sử dụng một biến Skolem để tính toán phần tử thiếu. Quá trình tính toán phần tử thiếu được mô tả theo thuật toán như sau:

Bước 1: Gán biến $k(a_1, a_2, \dots)$ đối với đoạn thiếu của truy vấn SG, ở đây k là một thực thể và (a_1, a_2, \dots) là tập thuộc tính của k

Bước 2: Xây dựng một đường dẫn đơn đối với mỗi liên kết có thể từ thực thể cuối và gán thực thể kể về ngữ nghĩa với thực thể cuối cho k .

Bước 3: Hiện thị tập các đường dẫn đã được tính cùng với truy vấn SG tương ứng và yêu cầu người dùng lựa chọn các đường dẫn mong muốn.

Bước 4: Xem đường dẫn đã chọn như truy vấn SG mới, lặp bước 2 và 3.

Bước 5: Lặp bước 4 cho đến khi toàn bộ đường dẫn mong muốn đã đạt được.

Bước 6: Dừng.

- Phân đoạn.

Mẫu SG của các truy vấn thuộc loại này bao gồm các đoạn rời nhau. Nhiệm vụ làm mịn truy vấn SG ở đây là tính toán các phần tử sơ đồ trung gian để nối các đoạn. Các kỹ thuật tương tự như trường hợp thiếu thông tin ở cuối.

- Phân đoạn và thiếu thông tin ở cuối.

Các truy vấn SG vừa bị phân đoạn vừa thiếu thông tin ở cuối trước tiên được xử lý để nối các đoạn rời nhau và thông tin thiếu ở cuối sẽ được giải quyết sau đó.

(2) Các truy vấn không rõ ràng.

- Các truy vấn không rõ ràng do khuyết tên thuộc tính và tên thực thể mà chỉ có giá trị sẽ được hệ thống tự động bổ sung thông qua các phép tra cứu chính CSDL đang được người dùng tra cứu.
- Tinh lược: Các câu truy vấn tinh lược là các câu không đầy đủ về cú pháp gồm có một mệnh đề ngữ cảnh mà là một câu truy vấn hoàn chỉnh và một mệnh đề tinh lược không hoàn chỉnh nhưng cấu trúc của nó phụ thuộc vào cấu trúc của mệnh đề ngữ cảnh. Ở đây, chúng tôi giả thiết rằng cấu trúc đầy đủ của mệnh đề tinh lược là giống với mệnh đề ngữ cảnh. Vấn đề sau đó chỉ là tìm sự tương ứng về cấu trúc giữa 2 mệnh đề này. Điều này đòi hỏi phải tách 2 mệnh đề trên, Chúng tôi sử dụng SG với một phép toán logic hay dấu (,) để tách 2 mệnh đề này. Tại mức SG, mệnh đề ngữ cảnh hoặc là một truy vấn SG đầy đủ hoặc là thiếu thông tin mà có thể được bổ sung như trên. Sau đó sử dụng mẫu SG và truy vấn SG của mệnh đề ngữ cảnh cho mệnh đề tinh lược.
- Đồng tham chiếu: Các truy vấn mà trong đó các cụm danh từ cùng tham chiếu đến các cụm danh từ khác được gọi là đồng tham chiếu. Các truy vấn loại này thường liên quan đến việc dùng đồng thời nhiều đại từ như: cô ấy, anh ấy, nó.. trong các câu ghép. Lần lượt, các đại từ sẽ được thay thế với các biến Skolem và gán cho các biến này một trong các phần tử có thể về ngữ nghĩa của truy vấn SG. Sau đó, hệ thống sẽ hiển thị cho người dùng các truy vấn SG này và yêu cầu người dùng lựa chọn cho tới khi tất cả các đại từ đều được xác định.

(3) Câu truy vấn nhập nhằng.

Câu truy vấn nhập nhằng là những câu đúng về cú pháp nhưng nhập nhằng về ngữ nghĩa. Không như những câu truy vấn tinh lược và câu truy vấn đồng tham chiếu, tùy thuộc vào từng loại câu truy vấn nhập nhằng mà ta có thể giải quyết bằng cách làm mịn câu truy vấn hoặc áp dụng các kỹ thuật như đã làm với câu truy vấn không rõ ràng.

5. THỬ NGHIỆM VÀ ĐÁNH GIÁ

Chúng tôi đã tiến hành cài đặt thử nghiệm một hệ truy vấn ngôn ngữ tự nhiên tiếng Việt đối với CSDL Quản lý học tập một khoá của trường Đại học Bách khoa. Sơ đồ thực thể-liên kết của CSDL thử nghiệm gồm các thực thể : SinhViên, GiảngViên, MônHọc, Lớp, Khoa và các liên kết : MônHọc --- Được-Dạy-Bởi → GiảngViên, SinhViên --- Được-HướngDẫn-Bởi → GiảngViên,

SinhViên --- Học → MônHọc,....

Ví dụ câu hỏi: Cho biết các sinh viên của giảng viên A. Truy vấn này sẽ được phân tích cú pháp và kết quả trả ra là:

Mẫu SG: Wh-Form◊TargetEntity◊
.....◊ Source Value

Truy vấn SG: Cho biết→SinhViên
...GiảngViên A

→ Đây là câu truy vấn bị phân đoạn

Bộ phân tích và xử lý ngữ nghĩa tham chiếu đến sơ đồ thực thể-liên kết, CSDL quan hệ để phân tích các mối quan hệ giữa các thực thể và xác định được 2 đường dẫn giữa SinhViên và GiảngViên là :

1. SinhViên --- Học → MônHọc --- Được-Dạy-Bởi → GiảngViên
2. SinhViên --- Được-HướngDẫn-Bởi → GiảngViên

Người sử dụng sẽ được hỏi để lựa chọn đường dẫn phù hợp. Theo lựa chọn của

người sử dụng, hệ sẽ xác định được truy vấn CQL tương ứng:

1.CQL = (SV; GV_{Tên=A}; SV Học MônHọc
Được-Dạy-Bồi GV)

2.CQL = (SV; GV_{Tên=A}; SV Được-
HướngDẫn-Bồi GV)

Bộ dịch CQL thành SQL sẽ đưa ra câu truy vấn SQL :

1.SELECT * FROM SV, Học, Dạy, GV
WHERE SV.MãSV = Học.MãSV AND
Học.MãM = Dạy.MãM AND Dạy.MãGV
= GV.MãGV AND TênGV = 'A'

2.SELECT * FROM SV, HướngDẫn, GV
WHERE SV.MãSV = HướngDẫn.MãSV
AND HướngDẫn.MãGV = GV.MãGV
AND TênGV = 'A'

So với cách tiếp cận sử dụng hệ logic mô tả [1], cách tiếp cận sử dụng SG cho phép xác định các kiểu truy vấn tự nhiên không hoàn chỉnh và kỹ thuật xử lý chúng một cách tường minh. Hơn nữa, cách tiếp cận này còn cho phép tương tác với người sử dụng và do vậy, người sử dụng có thể yên tâm về các xử lý truy vấn của hệ thống.

6. KẾT LUẬN

Hệ thống cài đặt đã đáp ứng được các yêu cầu và mục tiêu đề ra đối với một hệ thống truy vấn ngôn ngữ tự nhiên. Tuy nhiên, hiệu quả của hệ thống phụ thuộc rất nhiều vào vốn từ vựng mà ta đưa vào. Đây chính là khó khăn lớn nhất và cũng là vấn đề cơ bản của bất kỳ hệ thống xử lý ngôn ngữ tự nhiên nào- sự hiểu biết của nó về CSDL cụ thể.

Theo đánh giá của chúng tôi, hệ thống cài đặt đã đạt được các mục tiêu sau:

- Hệ thống đã cung cấp một giao diện khá thân thiện và thuận tiện cho người sử dụng khi truy vấn.
- Hệ thống có khả năng đưa ra những trợ giúp để giúp cho người sử dụng đặt câu hỏi đúng với mục đích.
- Với một số loại câu hỏi thì hệ thống có thể tự động sửa được.
- Hệ thống có tính khả chuyển cao. Khi thay đổi miền ứng dụng thì cần thay đổi các từ điển, sơ đồ thực thể liên kết và cơ sở dữ liệu quan hệ của hệ.

Cuối cùng, chúng tôi hy vọng rằng hệ thống cài đặt sẽ được cải tiến và phát triển hoàn thiện hơn nữa để đáp ứng đầy đủ các yêu cầu của một hệ truy vấn ngôn ngữ tự nhiên tiếng Việt và thực sự cho phép những người sử dụng không được đào tạo về Tin học có thể khai thác tốt các CSDL.

TÀI LIỆU THAM KHẢO

- [1] Nguyễn Kim Anh, Một cách tiếp cận đối với phép dịch các câu truy vấn ngôn ngữ tự nhiên thành dạng logic, Tạp chí Tin học và Điều khiển học, 21 (3), p. 271-279 (2005)
- [2] I. Androutsopoulos, Interfacing a Natural Language Front-End to Relational Database, Tech. Paper no.11, Dept.of AI, Univ. of Edingburgh (1993)
- [3] V.Owei, Enriching the conceptual basis for query formulation through relationship semantics in databases, Inf.Syst.26, 6, p.445-475 (2001)
- [4] V.Owei, An intelligent approach to handling imperfect information in concept-based natural language queries, ACM TOIS, 20(3), p.291-328 (2002)
- [5] D.L. Waltz, An English language question answering system for a large relational database, Comm. ACM, 21(7), p.526-539 (1978)