

ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
Phòng Đào tạo Sau đại học & Khoa học công nghệ

KHÓA 13 (ĐỢT 1)

BÁO CÁO HỆ CƠ SỞ TRI THỨC
**ỨNG DỤNG ONTOLOGY CHO HỆ
THỐNG TÌM KIẾM SÁCH THEO NGŨ
NGHĨA**

GIẢNG VIÊN HƯỚNG DẪN
PGS.TS. ĐỖ VĂN NHƠN

SINH VIÊN THỰC HIỆN
Lê Văn Duyệt
MSHV: CH1801004

TP. Hồ Chí Minh – Tháng 06/2019

MỤC LỤC

Mục lục	i
1 Tổng quan	1
1.1 Nhu cầu quản lý tài nguyên	1
1.2 Mục tiêu báo cáo	1
1.3 Phạm vi báo cáo	1
2 Cơ sở lý thuyết	2
2.1 Ontology	2
2.1.1 Ontology là gì?	2
2.1.2 Các phương pháp xây dựng Ontology	3
2.2 Hệ thống tìm kiếm thông tin	4
3 Mô hình giải pháp	5
3.1 Mô hình tổng thể	5
3.2 Mô hình Ontology biểu diễn ngữ nghĩa: CK-ONT	5
3.2.1 Tập hợp K các keyphrase	6
3.2.2 Tập hợp C các lớp keyphrase	6
3.2.3 Tập hợp R_{KC} các quan hệ giữa keyphrase và lớp	6
3.2.4 Tập hợp R_{CC} các quan hệ giữa các lớp	7
3.2.5 Tập hợp R_{KK} các quan hệ giữa các keyphrase	7
3.3 Tìm kiếm theo ngữ nghĩa	8
3.3.1 Công cụ xây dựng Ontology	8
3.3.2 Công cụ lập chỉ mục và tìm kiếm - Apache Solr	8
3.3.3 Quá trình thiết lập	10
4 Cài đặt	11
4.1 Yêu cầu và chức năng của hệ thống	11
4.2 Kiến trúc hệ thống	11
4.3 Cài đặt	12
4.3.1 Xây dựng Ontology	12
4.3.1.1 Thiết kế lớp	12
4.3.1.2 Thuộc tính lớp	13
4.3.1.3 Các mối quan hệ	13
4.3.2 Xây dựng các thành phần tạo chỉ mục	14
4.3.3 Xây dựng thành phần truy vấn	14

5	Kết luận	18
5.1	Kết quả đạt được	18
5.2	Hạn chế	18
5.3	Định hướng phát triển	18
	Tài liệu tham khảo	19

PHẦN 1: TỔNG QUAN

1.1 Nhu cầu quản lý tài nguyên

Nhu cầu quản lý tài nguyên tài liệu hướng dẫn là nhu cầu có thực và phổ biến hiện nay. Hiện nay việc tổ chức các kho tài nguyên như thế đã dần hoàn thiện và số lượng được lưu trữ của các tài liệu ngày càng được gia tăng. Với một số lượng khổng lồ các kho tài nguyên, việc tìm kiếm chính xác không phải là việc đơn giản. Việc tìm kiếm theo hướng ngữ nghĩa vẫn còn hạn chế. Việc tra cứu trước đây đến nay đa số dựa trên các siêu dữ liệu liên quan trong danh mục tài liệu mà ít có ứng dụng ngữ nghĩa vào việc tìm kiếm. Việc tìm kiếm không chính xác gây khó khăn cho người sử dụng và hiệu quả không cao, khai thác không được triệt để nguồn tài liệu thông tin.

Một vấn đề khác với các hệ thống tìm kiếm sách là ứng dụng một mô hình với mọi loại tài liệu thuộc nhiều lĩnh vực khác nhau. Trong khi có người tìm kiếm thường chỉ quan tâm đến một số thể loại nhất định tùy trường hợp. Việc sử dụng chung mô hình khó chính xác cao do đặc thù riêng, giảm hiệu quả đối với nhu cầu tìm kiếm thực tế.

1.2 Mục tiêu báo cáo

Xuất phát từ yêu cầu trên, báo cáo nhằm mục tiêu tìm hiểu xây dựng ứng dụng tìm kiếm cho hệ thống quản lý ebook trực tuyến. Đồng thời báo cáo tập trung vào một số thể loại sách cụ thể.

Công việc cụ thể:

- Có một hệ thống ebook online có sẵn, rút ra đặc thù riêng cho từng thể loại.
- Nghiên cứu phương pháp biểu diễn ngữ nghĩa của tài liệu để áp dụng vào quản lý kho ebook, mô hình Ontology mô tả tri thức lĩnh vực.
- Công cụ hỗ trợ xây dựng hệ thống tìm kiếm theo ngữ nghĩa.

1.3 Phạm vi báo cáo

Báo cáo chủ yếu tập trung tìm hiểu về sử dụng phương pháp Ontology vào xây dựng, tối ưu một hệ thống tìm kiếm có sẵn bằng cách bổ sung ngữ nghĩa, giúp cho việc tìm kiếm chính xác hơn. Lĩnh vực tìm kiếm là trên dữ liệu hệ thống quản lý chia sẻ ebook.

PHẦN 2: CƠ SỞ LÝ THUYẾT

2.1 Ontology

2.1.1 Ontology là gì?

Các cơ sở lý thuyết về Ontology được đề cập nhiều trong các tài liệu [1][2][3][4][5].

Thuật ngữ "Ontology" đã xuất hiện từ rất sớm. Trong cuốn sách "Siêu hình" Metaphysics[6]) của mình, Aristotle đã định nghĩa: "Ontology là một nhánh của triết học, liên quan đến sự tồn tại và bản chất các sự vật trong thực tế". Hay nói cách khác, đối tượng nghiên cứu chủ yếu của Ontology, xoay quanh việc phân loại các sự vật dựa trên các đặc điểm mang tính bản chất của nó. Ontology là một thuật ngữ mượn từ triết học được tạm dịch là "bản thể học", nhằm chỉ khoa học mô tả các loại thực thể trong thế giới thực và cách chúng liên kết với nhau.

Trong ngành khoa học máy tính và khoa học thông tin, Ontology mang ý nghĩa là các khái niệm lớp đối tượng và quan hệ giữa chúng trong một hệ thống hay ngữ cảnh cần quan tâm. Các khái niệm lớp đối tượng này còn được gọi là các khái niệm, các thuật ngữ hay các bộ từ vựng có thể được sử dụng trong một lĩnh vực chuyên môn nào đó. Ontology cũng có thể hiểu là một ngôn ngữ hay một tập các quy tắc được dùng để xây dựng một hệ thống Ontology. Một hệ thống Ontology định nghĩa một tập các từ vựng mang tính phổ biến trong lĩnh vực chuyên môn nào đó và các mối quan hệ giữa chúng. Sự định nghĩa này có thể được hiểu bởi cả con người lẫn máy tính. Một cách khái quát, có thể hiểu Ontology là một biểu diễn của sự khái niệm hoá thống nhất được chia sẻ của một miền tri thức hay một lĩnh vực nhất định. Nó cung cấp một bộ từ vựng chung bao gồm các khái niệm.

Các thuộc tính quan trọng và các định nghĩa về các khái niệm và các thuộc tính này. Ngoài bộ từ vựng, Ontology còn cung cấp các ràng buộc, đôi khi các ràng buộc này được coi như các giả định cơ sở về ý nghĩa mong muốn của bộ từ vựng, nó được sử dụng trong một lĩnh vực mà có thể được giao tiếp giữa người và các hệ thống ứng dụng phân tán khác.

Một Ontology bao gồm các thành phần như sau:

- Các cá thể (individuals): các thực thể hoặc các đối tượng.
- Các lớp (classes): các tập hợp, các bộ sưu tập, các khái niệm, các loại đối tượng hoặc các loại khác.
- Các thuộc tính (attributes): các khía cạnh, đặc tính, tính năng, đặc điểm hoặc các thông số mà các đối tượng và các lớp có thể có.

- Các quan hệ (relations): cách thức mà các lớp và các cá thể có thể liên kết với nhau.
- Các thuật ngữ chức năng (function terms): cấu trúc phức tạp được hình thành từ các mối quan hệ nhất định có thể được sử dụng thay cho một thuật ngữ cá thể trong một statement.
- Các hạn chế (restrictions): những mô tả chính thức được tuyên bố về những điều phải chính xác cho một số khẳng định được chấp nhận ở đầu vào.
- Các quy tắc (rules): một cặp nếu-thì (if-then) mô tả suy luận logic có thể được rút ra từ một khẳng định trong từng hình thức riêng.
- Các tiên đề (axioms): các khẳng định (bao gồm các quy tắc) trong một hình thức hợp lý với nhau bao gồm các lý thuyết tổng thể mà ontology mô tả trong lĩnh vực của ứng dụng.
- Các sự kiện (events): sự thay đổi các thuộc tính hoặc các mối quan hệ.

2.1.2 Các phương pháp xây dựng Ontology

Có nhiều phương pháp khác nhau để xây dựng một Ontology, nhưng nhìn chung các phương pháp đều thực hiện hai bước cơ bản là: xây dựng cấu trúc lớp phân cấp và định nghĩa các thuộc tính cho lớp. Trong thực tế, việc phát triển một Ontology để mô tả lĩnh vực cần quan tâm là một công việc không đơn giản, phụ thuộc rất nhiều vào công cụ sử dụng, tính chất, quy mô, sự thường xuyên biến đổi của miền cũng như các quan hệ phức tạp trong đó.

Những khó khăn này đòi hỏi công việc xây dựng Ontology phải là một quá trình lặp đi lặp lại, mỗi lần lặp cải thiện, tinh chế và phát triển dần sản phẩm chứ không phải là một quy trình khung với các công đoạn tách rời nhau. Công việc xây dựng Ontology cũng cần phải tính đến khả năng mở rộng lĩnh vực quan tâm trong tương lai, khả năng kế thừa các hệ thống Ontology có sẵn, cũng như tinh chỉnh để Ontology có khả năng mô tả tốt nhất các quan hệ phức tạp trong thế giới thực.

Một số nguyên tắc cơ bản của việc xây dựng Ontology thông qua các công đoạn sau đây:

- Xác định miền quan tâm và phạm vi của Ontology.

- Xem xét việc kế thừa các Ontology có sẵn.
- Liệt kê các thuật ngữ quan trọng trong Ontology.
- Xây dựng các lớp và cấu trúc lớp phân cấp.
- Định nghĩa các ràng buộc về thuộc tính và quan hệ của lớp.
- Tạo các thực thể cho lớp.

2.2 Hệ thống tìm kiếm thông tin

Mục tiêu của hệ thống tìm kiếm thông tin và tìm kiếm và đưa ra các thông tin liên quan nhất đến cho người dùng. Các hệ thống này có nhiệm vụ tổ chức, phân loại tài liệu và phục vụ tra cứu.

Cấu trúc của một hệ thống tìm kiếm thông tin:

- Lập chỉ mục (indexing): phân tích tài liệu nhằm xác định các chỉ mục biểu diễn nội dung của tài liệu. Có hai cách: (1) lập chỉ mục từ cấu trúc phân lớp có sẵn và (2) rút trích chỉ mục từ nội dung có trong kho tài liệu.
- Tra cứu (interrogation): hệ thống nhận yêu cầu từ người dùng thông qua câu truy vấn (query). Hệ thống tiến hành phân tích và biểu diễn sau đó qua một hàm so khớp để tìm ra tài liệu liên quan.

PHẦN 3: MÔ HÌNH GIẢI PHÁP

3.1 Mô hình tổng thể

Mô hình tổng thể cho kho dữ liệu ebook online sẽ bao gồm 4 thành phần chính:

$$(D, FS, DB, ONT) \quad (3.1)$$

Trong đó:

- *D (Document)* là tập các tài liệu được quản lý trong hệ thống. Bên cạnh nội dung của mình, mỗi tài liệu có một định danh riêng biệt, có các thông tin siêu dữ liệu liên quan và ngữ nghĩa của tài liệu được biểu diễn bằng một đồ thị key phrase.
- *FS (File system)* là hệ thống tập tin dùng để lưu trữ các chế bản điện tử của tài liệu trong hệ thống. Đây là thành phần cơ bản của một kho tài liệu điện tử và là cấp thấp nhất về mặt lưu trữ trong hệ thống.
- *DB (Database)* Cơ sở dữ liệu danh mục. Database lưu trữ thông tin siêu dữ liệu liên quan đến tài liệu cùng cấu trúc thư mục của hệ thống tập tin (File system). Đóng vai trò liên kết giữa File system với các thành phần trừu tượng khác trong hệ thống.
- *ONT (Ontology)* là mô hình ontology cho ngữ nghĩa của tài liệu, chi phối quá trình các thao tác xử lý liên quan đến ngữ nghĩa của tài liệu.

3.2 Mô hình Ontology biểu diễn ngữ nghĩa: CK-ONT

Mô hình được đề xuất để biểu diễn hệ thống tìm kiếm theo ngữ nghĩa dựa trên ontology là mô hình CK-ONT (Classed Keyphrase based Ontology). Mô hình gồm 6 thành phần.

$$(K, C, R_{KC}, R_{CC}, R_{KK}, label) \quad (3.2)$$

Trong phạm vi nghiên cứu, mô hình CK-OB được kế thừa từ mô hình CK-ONT và đưa vào sử dụng dùng để mô hình hóa cho các tài liệu là ebook lĩnh vực CNTT, mô hình gồm 5 thành phần như sau:

$$(K, C, R_{KC}, R_{CC}, R_{KK}) \quad (3.3)$$

Trong đó:

- K : tập hợp các Keyphrase thuộc lĩnh vực CNTT.
- C : tập hợp các lớp keyphrase
- R_{KC} : tập hợp các quan hệ giữa keyphrase và lớp.
- R_{CC} : tập hợp các quan hệ giữa các lớp.
- R_{KK} : tập hợp quan hệ giữa các keyphrase.

3.2.1 Tập hợp K các keyphrase

- Là thành phần chính hình thành nên các khái niệm của ontology.
- Keyphrase trong mô hình này là những cụm từ hay thuật ngữ chuyên ngành CNTT.
Ví dụ: "lập trình", "con trỏ", "bool", ...

3.2.2 Tập hợp C các lớp keyphrase

- Mỗi lớp keyphrase là một tập hợp các keyphrase có liên quan với nhau theo một tính chất hay ngữ nghĩa nào đó
- Một lớp keyphrase có thể chứa các keyphrase hoặc các lớp keyphrase.
- Khi một lớp có chứa các lớp khác sẽ tạo thành mối quan hệ phân cấp cha con.

$$C = \{c \in P(K) \mid c \text{ là lớp keyphrase mô tả lĩnh vực đang xét}\}$$

Ví dụ: Lớp $XuLyNN$ chứa các keyphrase liên quan đến xử lý ngôn ngữ tự nhiên như sau:

$$XuLyNN = \{\text{Tách từ, PosTag, TFIDF, phân lớp, Word2vec}\}$$

3.2.3 Tập hợp R_{KC} các quan hệ giữa keyphrase và lớp

Một quan hệ hai ngôi giữa K và C ($C \neq \emptyset, K \neq \emptyset$) là một tập con của $K \times C$ và $R_{KC} = \{r \mid r \subseteq K \times C\}$.

Trong phạm vi báo cáo này chỉ xét quan hệ *inClass* ("thuộc về") giữa keyphrase và lớp.

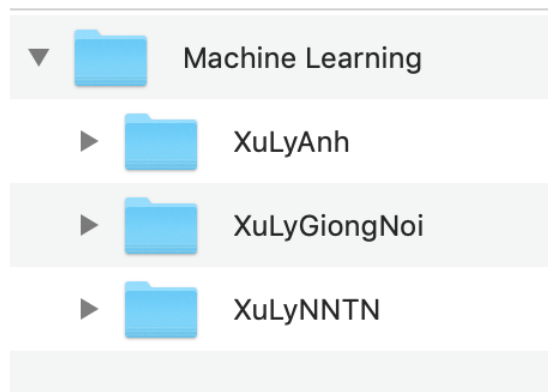
Ví dụ: "Tách từ" *inClass* $XuLyNN$, "TFIDF" *inClass* $XuLyNN$, ...

3.2.4 Tập hợp R_{CC} các quan hệ giữa các lớp

Một quan hệ hai ngôi trên tập hợp các lớp keyphrase C ($C \neq \emptyset$) là tập con của $C \times C$ và $R_{CC} = \{r \mid r \subseteq C \times C\}$.

Báo cáo chỉ xét quan hệ trên phân cấp lớp.

Ví dụ: Có sơ đồ phân cấp lớp như sau:



trong đó mỗi quan hệ giữa các lớp được mô tả như sau:

SuperClass	SubClass
Machine Learning	XuLyAnh
Machine Learning	XuLyGiongNoi
Machine Learning	XuLyNNTN

3.2.5 Tập hợp R_{KK} các quan hệ giữa các keyphrase

Một quan hệ hai ngôi trên K ($K \neq \emptyset$) là một tập hợp con của $K \times K$ nghĩa là một tập các cặp keyphrase thuộc K và $R_{KK} = \{r \mid r \subseteq K \times K\}$

Trong phạm vi đề tài ta chỉ xét các loại quan hệ sau:

	Quan hệ ngữ nghĩa	Mô tả
r1	Đồng nghĩa	A đồng nghĩa với B
r2	Viết tắt	A là dạng viết tắt của B
r3	Cùng lớp	A cùng lớp với B

- **Quan hệ đồng nghĩa r1:** hai keyphrase có quan hệ *đồng nghĩa* nếu chúng cùng nghĩa và thay thế được cho nhau trong một ngữ cảnh nào đó.

Ví dụ: keyphrase "công nghệ phần mềm" có quan hệ *đồng nghĩa* với keyphrase "kỹ thuật phần mềm"

- **Quan hệ Viết tắt r2:** hai keyphrase có quan hệ *viết tắt* nếu chúng cùng nghĩa với nhau và thay thế được cho nhau trong một ngữ cảnh nào đó.

Ví dụ: keyphrase "CSS"

- **Quan hệ Cùng lớp r3:** keyphrase a có quan hệ *cùng lớp* với keyphrase b nếu có một lớp C_i sao cho $a \in C_i$ và $b \in C_i$.

Ví dụ: keyphrase "Javascript" có quan hệ cùng lớp với keyphrase "Typescript".

3.3 Tìm kiếm theo ngữ nghĩa

3.3.1 Công cụ xây dựng Ontology

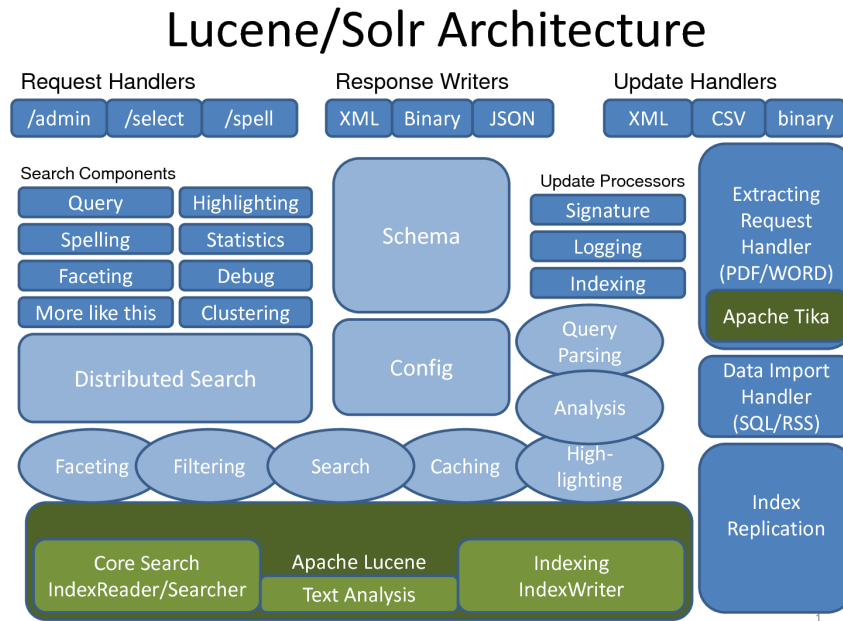
Protégé là công cụ phần mềm biên tập ontology mã nguồn mở (được phát triển tại Trường ĐH Stanford) sử dụng đối với việc xây dựng các hệ thống thông minh. Protégé được hỗ trợ bởi cộng đồng lớn bao gồm: các viện nghiên cứu, các tổ chức chính phủ và những người sử dụng cộng tác. Các đơn vị, cá nhân này sử dụng Protégé để xây dựng các giải pháp dựa trên tri thức trong các lĩnh vực chuyên sâu như là: y sinh học, thương mại điện tử và mô hình hóa tổ chức.

Chuẩn ngôn ngữ được sử dụng nhiều nhất để xây dựng ontology hiện nay là OWL[7] được phát triển bởi W3C. Giống như Protégé, OWL có thể mô tả các khái niệm nhưng nó cũng đưa ra các cách thức mới. Nó bao gồm tập rất nhiều các phép toán, ví dụ: phép giao (intersection), phép hợp (union) và phép phủ định (negation). Nó dựa trên một mô hình logic khác giúp nó có thể định nghĩa các khái niệm giống như cách mà các khái niệm đó đã được mô tả.

3.3.2 Công cụ lập chỉ mục và tìm kiếm - Apache Solr

Apache Solr là một nền tảng full-text search mã nguồn mở dựa trên Apache Lucene. Lucene là một thư viện được viết bằng Java dùng để phân tích, đánh chỉ mục (indexing) và tìm kiếm thông tin được phát triển đầu tiên bởi Doug Cutting vào năm 2000. Cutting đồng thời cũng là tác giả của Hadoop lúc ông đang làm việc cho Yahoo vào năm 2005. Solr không hoàn toàn là một RESTful interface của Lucene mà là sử dụng Lucene như là một component trong toàn bộ hệ thống.

Apache Solr có nhiều thành phần chính, nhưng quan trọng nhất là 2 thành phần: Thành phần tạo chỉ mục và thành phần tìm kiếm.



Hình 3.1: Các thành phần của Apache Solr

- **Thành phần Tạo chỉ mục:** bao gồm các thành phần hỗ trợ xử lý, tạo chỉ mục từ văn bản tài liệu đầu vào và cho ra kết quả là tập chỉ mục phục vụ cho thành phần tìm kiếm. Các component cơ bản bao gồm:
 - *Directory*: định nghĩa vùng nhớ, RAM nơi lưu chỉ mục.
 - *Document* và *Field*: định nghĩa tài liệu và các trường thông tin của tài liệu cho việc lập chỉ mục, dùng cho việc trả kết quả cho tìm kiếm.
 - *Analyzer*: thực hiện chức năng xử lý và tách văn bản để lấy nội dung, chuẩn hóa, loại bỏ từ không cần thiết, ... hỗ trợ cho bước lập chỉ mục.
Analyzer đóng vai trò khảo sát các trường văn bản để tạo ra một token stream. Ví dụ: *WhitespaceAnalyzer* xử lý phân tích văn bản thành những token dựa trên khoảng trắng. Câu "The quick brown fox jump over the lazy dog" sẽ được phân tích thành các tokens: [The] [quick] [brown] [fox] [jump] [over] [the] [lazy] [dog].
 - *Tokenizer* Nếu như Analyzer tạo ra các token streams/input stream, thì Tokenizer chia nhỏ các stream đó thành những tokens (đơn vị nhỏ nhất để

index, có thể là từ hay ký tự). Các ký tự trong input stream có thể bị bỏ qua như các ký tự không nhìn thấy được (whitespace như khoảng trắng, tab) hay các dấu phân cách (delimiter như dấu phẩy, dấu chấm).

- *IndexWriter* thực hiện việc tạo mới, mở chỉ mục, thêm hoặc cập nhật nội dung chỉ mục. Đây là thành phần chính trong thành phần tạo chỉ mục.

- **Thành phần Tìm kiếm** bao gồm các thành phần chức năng phục vụ cho việc xử lý tìm kiếm từ yêu cầu người dùng. Solr hỗ trợ nhiều loại truy vấn khác nhau, cho phép tìm theo trường thông tin hay các thiết lập nâng cao như sắp xếp kết quả, giới hạn thời gian hoặc số lượng kết quả, ...

Các chức năng cơ bản gồm:

- *Term*: là đơn vị cơ bản của tìm kiếm, gồm tên và giá trị tương ứng.
- *Query* gồm nhiều loại truy vấn, chứa nhiều phương thức, thiết lập chỉ số Boost nhằm giúp xác định truy vấn con nào quan trọng hơn.
- *IndexSearcher* tìm kiếm trên tập chỉ mục *IndexWriter* tạo ra.

3.3.3 Quá trình thiết lập

- Xây dựng tập chỉ mục tìm kiếm
 - Mô hình hóa nội dung với Apache Solr
 - Tiến trình lập chỉ mục
- Tìm kiếm trên tập chỉ mục

PHẦN 4: CÀI ĐẶT

4.1 Yêu cầu và chức năng của hệ thống

Yêu cầu hệ thống tìm kiếm:

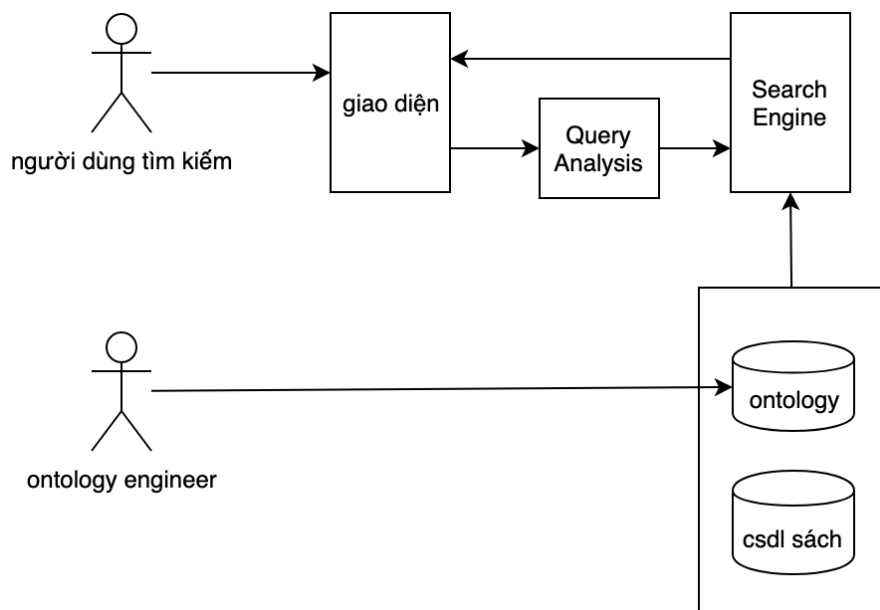
- Hỗ trợ tìm kiếm theo chức năng thông thường và tìm kiếm theo ngữ nghĩa.
- Có một ontology mô tả tri thức lĩnh vực sách CNTT.
- Kết quả đáp ứng được nhu cầu tìm kiếm của người dùng.

Hệ thống cho phép tìm kiếm theo các cách thức sau:

1. Tìm kiếm so trùng dựa vào mọi từ người dùng nhập vào. Kết quả trả về bao gồm tài liệu có các thành phần sau chứa từ khóa tìm kiếm: tiêu đề sách, tên tác giả, tập từ khóa tài liệu.
2. Tìm kiếm không so trùng chính xác tuyệt đối từ khóa người dùng nhập vào. Hệ thống sẽ tách câu, tách từ, chọn lọc bổ sung các từ khóa sử dụng ontology. Sau đó dùng các từ khóa này để so trùng như cách 1.

4.2 Kiến trúc hệ thống

Kiến trúc hệ thống như sau:



Hình 4.1: Kiến trúc hệ thống tìm kiếm ngữ nghĩa với Ontology

Mô tả các thành phần trong hệ thống:

- **Database:** cơ sở dữ liệu của toàn hệ thống bao gồm *CSDL SACH* lưu trữ thông tin sách cho hệ thống và *CSDL ONTOLOGY* là ontology cho sách lĩnh vực CNTT.
- **Giao diện:** giao tiếp giữa người dùng và hệ thống.
- **Query Analysis:** tiếp nhận thông tin từ **Giao diện** chuẩn hóa thông tin sau đó đưa thông tin đến **Search Engine** của hệ thống.
- **Search Engine:** bộ tìm kiếm sẽ nhận thông tin theo cấu trúc đặc tả từ Query Analysis sau đó thực hiện tìm kiếm, trả kết quả về Giao diện.

4.3 Cài đặt

Các bước cài đặt bao gồm:

- Xây dựng Ontology cho ứng dụng.
- Xây dựng thành phần tạo chỉ mục.
- Xây dựng thành phần xử lý câu truy vấn và truy vấn dữ liệu dựa trên yêu cầu truy vấn người dùng.

4.3.1 Xây dựng Ontology

Các khái niệm được sử dụng trong cơ sở tri thức Ebook là những khái niệm thuộc lĩnh vực giáo dục, được trích ra từ các sách thu thập được trên hệ thống. Báo cáo chỉ tập trung phục vụ tra cứu cho một nhóm nhỏ sách ebook thuộc lĩnh vực CNTT, nên sẽ xây dựng ontology dựa trên dữ liệu sách này.

Ontology trong báo cáo này được rút trích từ mục lục các quyển sách phổ biến trên hệ thống. Các công việc bao gồm thực hiện việc phân loại và rút trích keyphrase từ mục lục, công việc được thực hiện một cách thủ công.

4.3.1.1 Thiết kế lớp

Từ các dữ liệu thu thập được, đề tài đưa ra 7 lớp như sau:

1. *TacGia*: lớp tổng quan về tác giả.

2. *Sach*: lớp tổng quan về sách.
3. *Keyphrase*: lớp tổng quan về Keyphrase. Gồm các lớp con sau:
4. *CNTT*: lớp các keyphrase về sách chuyên ngành công nghệ thông tin.
5. *ML*: lớp các keyphrase về sách chuyên ngành trí tuệ nhân tạo Machine Learning.
6. *ThietKeWeb*: lớp các keyphrase về sách thiết kế Website.
7. *CSDL*: lớp các keyphrase về sách cơ sở dữ liệu.

4.3.1.2 Thuộc tính lớp

Các thuộc tính lớp được xây dựng cho ứng dụng dựa trên chuẩn từ vựng Dublin Core và được bổ sung thêm:

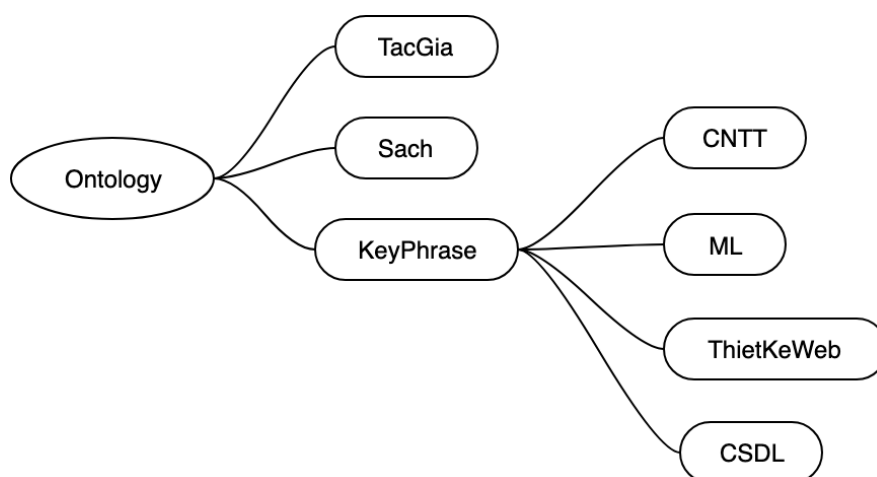
- *ma_sach*: mã sách trên hệ thống.
- *tieu_de*: tiêu đề sách ebook.
- *ten_tg*: tiêu tác giả.
- *key_phrase*: danh sách keyphrase biểu diễn nội dung sách.

4.3.1.3 Các mối quan hệ

Xây dựng các mối quan hệ bao gồm:

- Quan hệ liên quan giữa các lớp:
 - Quan hệ giữa Sách và Tác giả: Sách **coTacGia** TacGia
 - Quan hệ giữa Tác giả và Sách: TacGia **laTacGiaCua** Sach
 - Quan hệ giữa Sách và Tác Giả Phụ: TacGia **coTacGiaPhu** Sach
 - Quan hệ giữa Sách và Keyphrase: Sách **coKeyphrase** Keyphrase
 - Quan hệ giữa Keyphrase và Sách: Keyphrase **coKeyphrase** Sach.
- Quan hệ giữa các keyphrase:
 - Quan hệ đồng nghĩa

- Quan hệ viết tắt
- Quan hệ cùng lớp
- Quan hệ phân cấp trên lớp



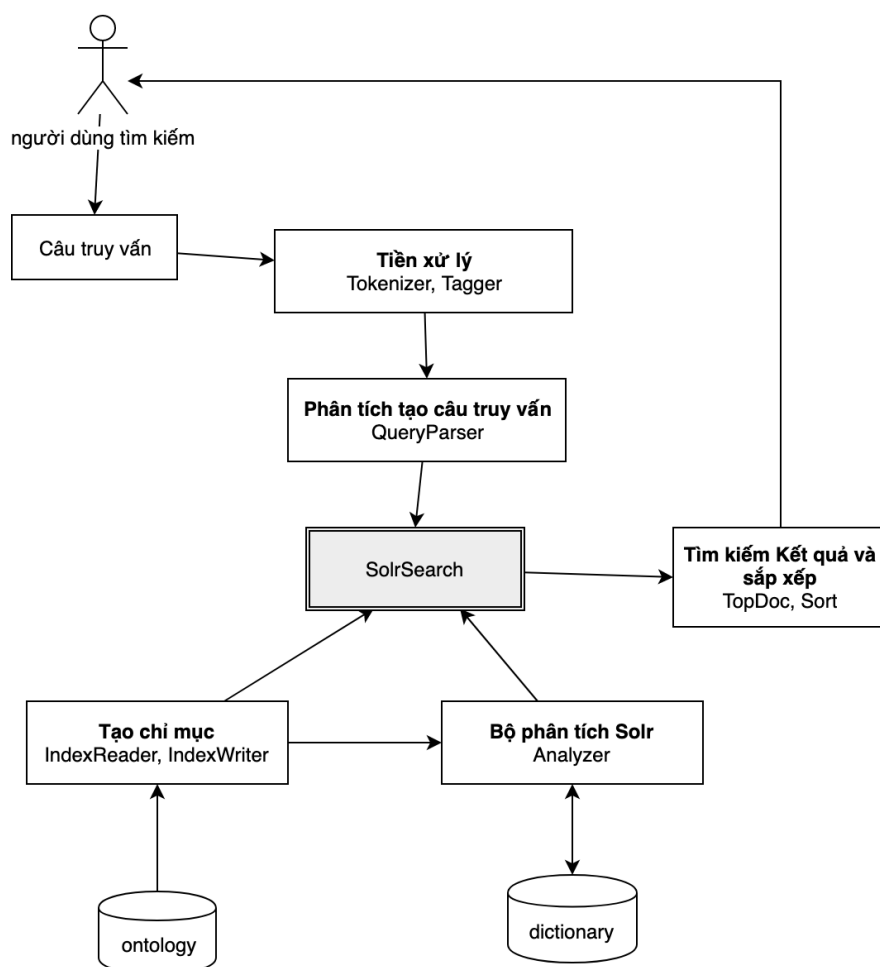
Hình 4.2: Minh họa các quan hệ phân cấp trên lớp

4.3.2 Xây dựng các thành phần tạo chỉ mục

Thành phần tạo chỉ mục bao gồm các chức năng chính như chỉ định dữ liệu lập chỉ mục, thực hiện phân tích tài liệu, tạo chỉ mục và lưu trữ. Thành phần này kế thừa từ thư viện Apache Solr và Lucene.

4.3.3 Xây dựng thành phần truy vấn

Thành phần truy vấn gồm các chức năng chính như: nhận thông tin truy vấn, chuyển đổi từ truy vấn và tìm kiếm, hiển thị kết quả trả về. Thành phần biên dịch truy vấn và tìm kiếm cũng kế thừa từ Apache Solr. Quy trình tổng quát như hình 4.3.



Hình 4.3: Quy trình xử lý trên hệ thống tìm kiếm

- Chuẩn bị dữ liệu cho hệ thống tìm kiếm thực hiện qua các bước

1. Đọc file ontology
2. Tạo cây dữ liệu từ ontology: *CreateNodes*
3. Tạo chỉ mục tìm kiếm: *IndexReader, IndexWriter*

- Quy trình xử lý tìm kiếm hiện qua các bước

1. Người dùng nhập câu
2. Tiền xử lý: tách từ, gán nhãn cho câu truy vấn
3. Xây dựng cây truy vấn theo chuẩn Apache Solr: *QueryParser*
4. Thực hiện tìm kiếm: *SolrSearch*

5. Đánh giá kết quả và sắp xếp theo độ đo: *TopDoc*, *TopDoc.Sort*
6. Hiển thị kết quả cho người dùng.

- **Thuật toán**

- Thuật toán xây dựng cấu trúc tìm kiếm từ file Ontology

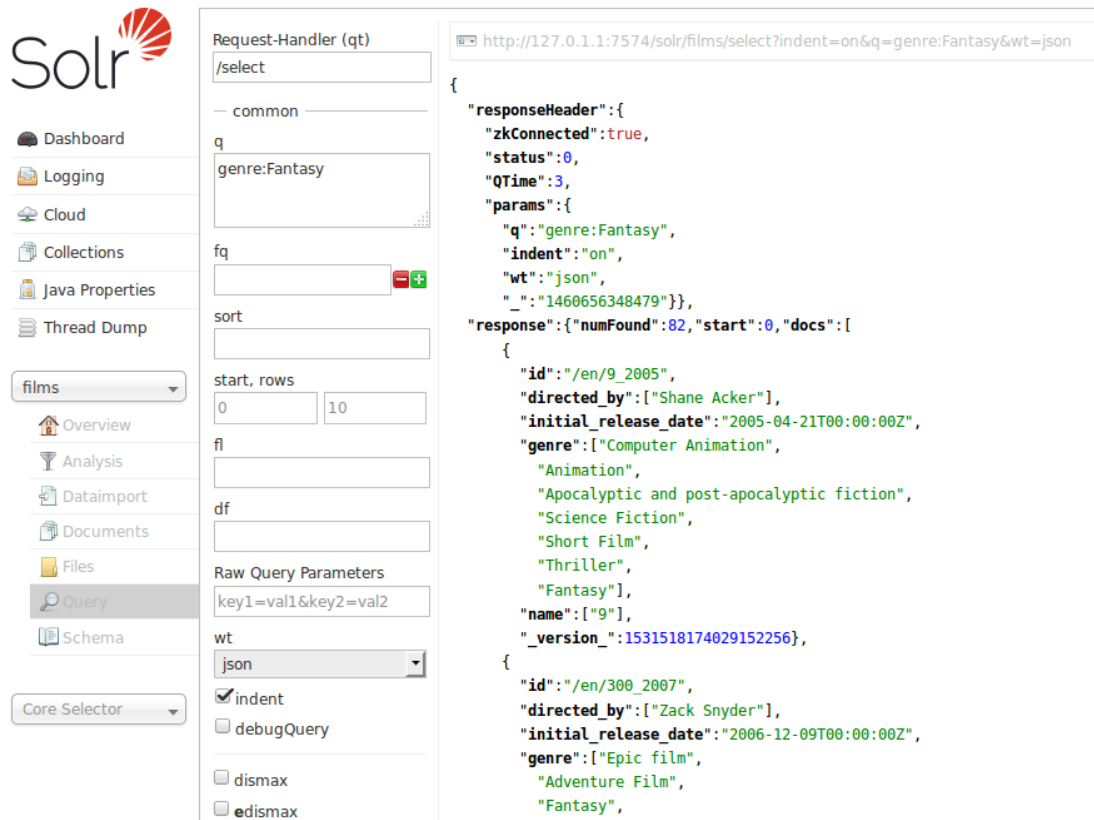
```
1 loadOntology(iw IndexWriter, o Sach):  
2     iw.MaSach = getMaSach(s)  
3     iw.TuaSach = getTuaSach(s)  
4  
5     while KeyPhrase in getKeyPhrase(s):  
6         iw.KeyPhase[] = KeyPhrase  
7  
8     while TacGia in getTacGia(s):  
9         iw.TacGia[] = TacGia
```

- Lấy toàn bộ thông tin sách và load vào Solr Index

```
1 buildIndex():  
2     // Khoi tao IndexWriter  
3     IndexWrite iw = IndexWrite{}  
4  
5     // Lay danh sach ebooks tu Database  
6     Sach[] books = getBooksFromDB()  
7  
8     // Load vao index Solr  
9     while book in books:  
10         loadOntology(iw, book)
```

- **Giao diện tìm kiếm Apache Solr**

Hình 4.4 thể hiện giao diện tìm kiếm của Apache Solr Client dưới dạng web, kết quả trả về có thể ở dạng JSON hoặc XML, dễ dàng tích hợp vào các hệ thống Web hoặc Application khác.



Solr

- Dashboard
- Logging
- Cloud
- Collections
- Java Properties
- Thread Dump
- films
 - Overview
 - Analysis
 - Dataimport
 - Documents
 - Files
 - Query
 - Schema
- Core Selector

Request-Handler (qt)

/select

common

q
genre:Fantasy

fq

sort

start, rows
0 10

fl

df

Raw Query Parameters
key1=val1&key2=val2

wt
json

☒ indent

☐ debugQuery

☐ dismax

☐ edismax

http://127.0.1.1:7574/solr/films/select?indent=on&q=genre:Fantasy&wt=json

```
{
  "responseHeader":{
    "zkConnected":true,
    "status":0,
    "QTime":3,
    "params":{
      "q":"genre:Fantasy",
      "indent":"on",
      "wt":"json",
      "_":"1460656348479"}},
  "response":{"numFound":82,"start":0,"docs":[
    {
      "id":"/en/9_2005",
      "directed_by":["Shane Acker"],
      "initial_release_date":"2005-04-21T00:00:00Z",
      "genre":["Computer Animation",
        "Animation",
        "Apocalyptic and post-apocalyptic fiction",
        "Science Fiction",
        "Short Film",
        "Thriller",
        "Fantasy"],
      "name":["9"],
      "_version_":1531518174029152256},
    {
      "id":"/en/300_2007",
      "directed_by":["Zack Snyder"],
      "initial_release_date":"2006-12-09T00:00:00Z",
      "genre":["Epic film",
        "Adventure Film",
        "Fantasy",
```

Hình 4.4: Giao diện tìm kiếm Apache Solr Client

PHẦN 5: KẾT LUẬN

5.1 Kết quả đạt được

Qua kết quả ta thấy tầm quan trọng của việc biểu diễn tri thức và ứng dụng của nó trong thực tiễn, cụ thể trong báo cáo này giúp tăng độ chính xác cho hệ thống tìm kiếm bằng cách bổ sung ngữ nghĩa. Báo cáo có những kết quả như sau:

- Hiểu cách xây dựng một ontology từ dữ liệu sách ebook.
- Ứng dụng ontology vào thư viện chỉ mục và tìm kiếm Apache Solr.
- Tìm kiếm tài liệu bằng cách chọn lọc lại từ khóa trong câu truy vấn của người dùng.

5.2 Hạn chế

Thời gian có hạn nên báo cáo còn khá nhiều chức năng chưa thực hiện:

- Chưa cho phép truy cập, hiệu chỉnh Ontology
- Chưa tập hợp được kiến thức chuyên gia để xây dựng hệ Ontology.
- Chỉ tập trung xây dựng ứng dụng một lĩnh vực cụ thể.
- Chỉ kiểm tra trên những câu truy vấn cơ bản, phụ thuộc vào các tập keyphrase đã được xây dựng sẵn trong ontology.

5.3 Định hướng phát triển

Để phát triển và đưa vào thực tiễn, cần:

- Xây dựng hoàn thiện Ontology từ chuyên gia.
- Mở rộng sang các miền, chuyên ngành sách khác.

TÀI LIỆU THAM KHẢO

- [1] Deborah L McGuinness, Frank Van Harmelen, et al. Owl web ontology language overview. *W3C recommendation*, 10(10):2004, 2004. [2](#)
- [2] Alexander Maedche and Steffen Staab. Ontology learning for the semantic web. *IEEE Intelligent systems*, 16(2):72–79, 2001. [2](#)
- [3] Tim Berners-Lee, James Hendler, Ora Lassila, et al. The semantic web. *Scientific american*, 284(5):28–37, 2001. [2](#)
- [4] Hồ Trung Thành and Đỗ Phúc. Ontology tiếng việt trong lĩnh vực giáo dục đại học. *Tạp chí Khoa học Công nghệ, Viện Hàn lâm Khoa học Công nghệ Việt Nam, Tập*, 52:89–100, 2014. [2](#)
- [5] Hoàng Văn Kiêm and Đỗ Văn Nhơn. Mạng tính toán và ứng dụng. *Journal of Computer Science and Cybernetics*, 13(3):10–20, 1997. [2](#)
- [6] Richard Taylor and Edward Seago. *Metaphysics*. Prentice-Hall Englewood Cliffs, 1963. [2](#)
- [7] Grigoris Antoniou and Frank Van Harmelen. Web ontology language: Owl. In *Handbook on ontologies*, pages 67–92. Springer, 2004. [8](#)