HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

TRẦN THỊ TUYẾT HOA

XÂY DỰNG HỆ THỐNG TRUY HỒI HỌC LIỆU CHO SINH VIÊN NGÀNH ĐIỆN - ĐIỆN TỬ

LUẬN VĂN THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

TP.HỒ CHÍ MINH - NĂM 2022

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

TRẦN THỊ TUYẾT HOA

XÂY DỰNG HỆ THỐNG TRUY HỒI HỌC LIỆU CHO SINH VIÊN NGÀNH ĐIỆN - ĐIỆN TỬ

CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN

MÃ SỐ: 8.48.01.04

LUẬN VĂN THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC

TS. TÂN HẠNH

TP.HỒ CHÍ MINH - NĂM 2022

LỜI CẢM ƠN

Trước tiên, em xin gửi lời cảm ơn chân thành đến quí Thầy Cô của Học viện Công Nghệ Bưu Chính Viễn thông cơ sở tại TP.HCM đã truyền đạt những kiến thức quý báo cho em trong suốt thời gian học tập vừa qua. Đặc biệt, em xin gửi lời cảmơn sâu sắc đến Thầy **TS. Tân Hạnh** đã tận tình hướng dẫn, giảng dạy em trong quá trình học tập cũng như hoàn thành luận văn tốt nghiệp.

Sau cùng, em xin cảm ơn gia đình, bạn bè và đồng nghiệp đã động viên, chia sẻ và tạo điều kiện cho em hoàn thành luận văn này.

Tuy có nhiều cố gắng trong quá trình học tập, cũng như quá trình hoàn thành luận văn tốt nghiệp không thể tránh khỏi những thiếu sót, em rất mong được sự góp ý quý báu của tất cả của quí Thầy Cô cũng như tất cả các anh chị để kết quả của em được hoàn thiện hơn.

Xin kính chúc quí Thầy Cô nhiều sức khỏe, thành công và hạnh phúc phúc. Em xin chân thành cảm ơn.

TP.HCM, ngày 15 tháng 07 năm 2022

Học viên thực hiện luận văn

Trần Thị Tuyết Hoa

LỜI CAM ĐOAN

Tôi xin cam đoan luận văn thạc sĩ chuyên ngành hệ thống thông tin **"Xây dựng hệ thống truy hồi học liệu cho sinh viên ngành điện – điện tử"** là do tôi nghiên cứu, tổng hợp và thực hiện dưới sự hướng dẫn của Thầy **TS. Tân Hạnh**.

Toàn bộ luận văn, những nội dung trình bày là của chính cá nhân tôi hoặc là được tham khảo, tổng hợp từ nhiều nguồn tài liệu khác nhau. Tất cả các tài liệu tham khảo, tổng hợp đều được trích xuất nguồn gốc rõ ràng. Các số liệu, kết quả được nêu trong luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

TP.HCM, ngày 15 tháng 07 năm 2022

Học viên thực hiện luận văn

Trần Thị Tuyết Hoa

MỤC LỤC

DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
IR	Information Retrieval	Truy hồi thông tin
IRS	Information Retrieval Systems	Hệ thống tìm kiếm thông tin
AI	Artificial Intelligence	Trí tuệ nhân tạo
NPL	Natural Language Processing	Xử lý ngôn ngữ tự nhiên
ID	Identification	Nhận dạng
TF	Term Frequency	Tần suất xuất hiện của thuật ngữ
IDF	Inverse Document Frequency	Tần suất nghịch đảo văn bản
D	Document collection	Tập hợp tài liệu
Q	Query collection	Tập hợp truy vấn
F	Framework	Mô hình mô tả tài liệu
R	Ranking function	Hàm xếp hạng
PDF	Portable Document Format	
HTML	Hypertext Markup Language	
UI	User Interface	Giao diện người dùng
P	Precision	Độ chính xác
R	Recall	Độ bao phủ

DANH SÁCH BẢNG

DANH SÁCH HÌNH VẼ

MỞ ĐẦU

1. Lý do chọn đề tài

Hiện nay, với sự phát triển không ngừng của ngành công nghệ thông tin nên nhu cầu tìm kiếm dữ liệu, tài liệu môn học của sinh viên cũng được phát triển theo. Một sinh viên không cần phải mất nhiều thời gian và công sức đi đến thư viện của trường học để tìm kiếm tài liệu các môn học như trước đây nữa, mà cách tìm kiếm đã được thay đổi hoàn toàn nhanh chóng hơn chỉ trong vài cái click chuột trên bàn phím máy vi tính thông qua Google hay các địa chỉ Web thông dụng.

Bên cạnh đó trong bối cảnh Thế giới hiện tại, thì tình hình dịch Covid đang là vấn đề đáng quan tâm và lo ngại nên môi trường học tập của sinh viên – học sinh dần chuyển sang hình thức học trực tuyến đã được phổ biến rộng khắp các trường học ở Việt Nam. Với hình thức học trực tuyến – online thì nhu cầu tìm kiếm thông tin, tài liệu chính xác về môn học cũng được nâng cao theo. Khi một học sinh tìm lại liệu trên Web mạng thì sẽ có hàng loạt các nội dung liên quan có khi đúng, gần đúng, khi không chính xác hoặc có nhiều địa chỉ truy cập xấu hiện ra. Lý do là vì phải đối mặt với nhiều nguồn tài liệu, hệ thống truy hồi thông tin dựa vào từ khóa sẽ không mang đến kết quả chính xác chuẩn như mong muốn.

Trước vấn đề đó, bản thân là một giáo viên trong ngành điện – điện tử tại trường Trung cấp Kinh tế - kỹ thuật Tây Ninh mong muốn xây dựng một hệ thống truy hồi cho sinh viên trong ngành truy nhập tìm kiếm học liệu một cách hiệu quả chính xác nhất thông qua nguồn thư viện tài liệu tin cậy tại trường.

Từ mong muốn và ý tưởng trên tôi cũng sự ủng hộ và đồng ý hướng dẫn của TS. Tân Hạnh tôi chọn đề tài luận văn: **“Xây dựng hệ thống truy hồi học liệu cho sinh viên ngành điện – điện tử”**, luận văn hoàn thành sẽ góp phần vào việc giải quyết các vấn đề cần thiết cấp bách trong thực tế.

2. Tổng quan về vấn đề nghiên cứu

Đề tài hướng đến xây dựng và áp dụng có hiệu quả hệ thống tìm kiếm học liệucho sinh viên ngành điện – điện tử nhằm hỗ trợ kịp thời cho sinh viên làm tài liệu tham khảo học trực tuyến trong tình hình chung căn cứ theo chỉ đạo của thủ tướng chính phủ về việc giãn cách xã hội và kế hoạch đào tạo cần phải hoàn thành của trường.

Để thực hiện được mục tiêu ý tưởng đề ra, đề tài cần phải nghiên cứu và tiến hànhcác nội dung sau:

- Tìm hiểu và phân tích nhu cầu các nội dung môn học liên quan đến ngành điện – điện tử.
- Nghiên cứu từ cơ sở lý thuyết về hệ thống tìm kiếm, truy hồi thông tin, phân loại thể loại văn bản tìm kiếm. Từ đó áp dụng làm nền tảng để xây dựng và triển khai ứng dụng hệ thống tìm kiếm thông tin.

3. Mục tiêu nghiên cứu

- Mục tiêu chính: xây dựng hệ thống truy hồi học liệu cho sinh viên ngành điện – điện tử.
- Mục tiêu cụ thể của hệ thống gồm có các chức năng:
  - Truy hồi thông tin theo từ khóa
  - Truy hồi thông tin theo từ khóa và loại văn bản
  - Chức năng học và phân loại văn bản theo thể loại học liệu
  - Chức năng học và phân loại theo chủ đề thuộc lĩnh vực điện - điện tử.

4. Đối tượng và phạm vi nghiên cứu

4.1. Đối tượng:

- Các lý thuyết về truy hồi thông tin (Information Retrieval - IR)
- Hệ thống tìm kiếm thông tin (Information Retrieval Systems- IRS)
- Nghiên cứu về các quá trình truy hồi thông tin, các hướng tiếp cận giải quyết bài toán về truy hồi thông tin.

- Nghiên cứu về phân loại ngữ nghĩa văn bản tự động dựa trên kỹ thuật máy học (machine learning techniques)
- Phân tích, khảo sát và xây dựng hệ thống truy hồi học liệu cho sinh viên ngành điện – điện tử tại trường trung cấp kinh tế kỹ thuật Tây Ninh.

4.2 Phạm vi:

- Học liệu thuộc ngành điện – điện tử
- Ngôn ngữ tiếng Anh, Việt
- Thể loại học liệu: giáo trình, sách tham khảo

5. Phương pháp nghiên cứu

- Phương pháp phân tích và tổng hợp lý thuyết về truy hồi thông tin, phân loại văn bản dựa trên học máy.
- Phương pháp thực nghiệm khoa học: Xây dựng mô hình ứng dụng nhằm đánh giá hiệu quả của giải pháp và đánh giá kết quả thực nghiệm.

Chương 1: TỔNG QUAN VỀ TRUY HỒI THÔNG TIN

Hiện nay, truy hồi thông tin là vấn đề khá phổ biến trên toàn Thế giới. Nó đã, đang và sẽ được ứng dụng rộng rãi trong tất cả các lĩnh vực khi có nhu cầu tìm kiếm và truy hồi thông tin. Qua quá trình tìm hiểu từ những cơ sở lý thuyết đến thực tiễn về truy hồi thông tin, tôi đã xây dựng một giải pháp cơ bản nhằm giải quyết yêu cầu bài toán đặt ra với trình tự các phương pháp sau.

Tổng quan về quá trình truy hồi thông tin, chia làm 2 giai đoạn:

1. Giai đoạn tiền xử lý.

+ Xử lý ngôn ngữ tự nhiên.

+ Chỉ mục và đánh trọng số thuật ngữ liên quan.

2. Giai đoạn thu thập.

+ Xử lý truy vấn ứng dụng mô hình Boolean

+ Tìm kiếm thông qua thuật ngữ liên quan có trong chỉ mục.

+ Xếp hạng thứ tự liên quan của tài liệu trả về.

+ Phản hồi độ liên quan: dùng công thức tính độ chính xác (Precision) và độ bao phủ (Recall) đánh giá mức độ liên quan của tài liệu trả về.

Bên cạnh các giải pháp trên, tôi đã áp dụng phần mềm mã nguồn mở Lucene để thực hiện giải quyết bài toán đưa ra một cách hiệu quả nhất.

Nội dung chương 1 là giới thiệu sơ lược về truy hồi thông tin. Các nội dung được đề cập đến như sau:

- Khái niệm về truy hồi thông tin
- Các giai đoạn trong quá trình truy hồi thông tin
- Giới thiệu phần mềm mã nguồn mở Lucene
- Các phương pháp giải quyết truy hồi thông tin.
- Đánh giá hiệu quả của việc truy hồi thông tin

1.1. Các khái niệm truy hồi thông tin

Thuật ngữ truy hồi thông tin (Information Retrieval - IR), là việc tìm kiếm tài liệu ở trạng thái phi cấu trúc (thường là văn bản) đáp ứng nhu cầu thông tin nhất định từ các tập tin lớn trên máy tính, máy chủ cục bộ hoặc trên Internet [1].

IR là lĩnh vực khoa học máy tính chuyên về lý thuyết và thực hành tìm kiếm thông tin. Vì văn bản là phương tiện phổ biến nhất được sử dụng để biển diễn và phân phối thông tin một cách hiệu quả, nên hầu hết các nghiên cứu về IR đều tập trung vào việc tìm kiếm thông qua các bộ sưu tập văn bản của tài liệu [25].

Việc truy hồi thông tin có thể có nhiều hình thức khác nhau. Người dùng có thể bày tỏ nhu cầu thông tin của họ dưới dạng một truy vấn văn bản — bằng cách gõ trên bàn phím, bằng cách chọn đề xuất truy vấn hoặc bằng giọng nói nhận dạng — hoặc truy vấn có thể ở dạng hình ảnh, hoặc một số các trường hợp nhu cầu có thể được ngầm hiểu. Việc truy hồi có thể liên quan đến việc xếp hạng hiện có các phần nội dung, chẳng hạn như tài liệu hoặc câu trả lời ngắn, hoặc sáng tác phản hồi mới kết hợp thông tin đã truy hồi. Cả hai thông tin nhu cầu và các kết quả được truy hồi có thể sử dụng cùng một phương thức (*ví dụ:* truy hồi tài liệu văn bản để đáp ứng với truy vấn từ khóa), hoặc là khác nhau (*ví dụ*, tìm kiếm hình ảnh bằng cách sử dụng truy vấn văn bản).

Nếu truy vấn không rõ ràng, hệ thống truy hồi có thể xem xét lịch sử người dùng, vị trí thực tế, các thay đổi theo thời gian trong thông tin, hoặc ngữ cảnh khác khi xếp hạng kết quả. Hệ thống IR có thể cũng giúp người dùng hình thành ý định của họ (*ví dụ:* thông qua tự động hoàn thành truy vấn hoặc gợi ý truy vấn) và có thể trích xuất tóm tắt ngắn gọn các kết quả xem xét truy vấn của người dùng.

Một truy vấn tìm kiếm thường có thể chứa một vài thuật ngữ, trong khi tài liệu - đề cập đến độ dài, tùy thuộc vào tình huống, có thể dao động từ một vài thuật ngữ đến hàng trăm câu hoặc hơn. Mô hình Neural đại diện cho véc-tơ sử dụng IR gửi lại văn bản và thường chứa một số lượng lớn các tham số cần được điều chỉnh.

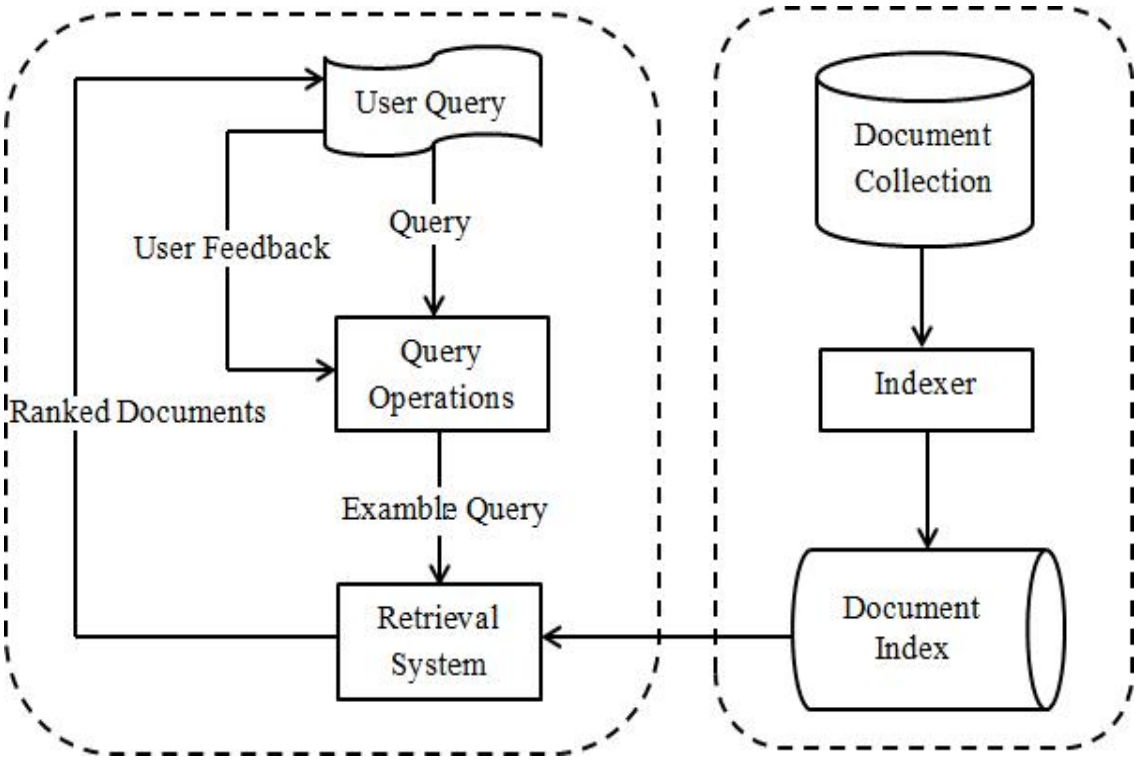
Trong các lĩnh vực khác, việc thiết kế các mô hình mạng nơ-ron đã được hình thành bởi các đặc điểm của ứng dụng và dữ liệu. Ví dụ, các bộ dữ liệu và các kiến trúc thành công khá khác nhau về hình ảnh nhận dạng đối tượng, nhận dạng giọng nói và tác nhân chính. Trong khi IR chia sẻ một số thuộc tính chung với lĩnh vực ngôn ngữ tự nhiên trong quá trình xử lý, nó cũng đi kèm với những thách thức riêng. Hệ thống IR-tems phải xử lý các truy vấn ngắn có thể chứa từ vựng, để so khớp với các tài liệu có độ dài khác nhau, để tìm các tài liệu liên quan cũng có thể chứa các phần lớn không liên quan chữ. Hệ thống IR nên học các mẫu trong văn bản truy vấn và tài liệu cho biết mức độ liên quan, ngay cả khi truy vấn và tài liệu sử dụng các từ vựng khác nhau, và ngay cả khi các mẫu là nhiệm vụ cụ thể hoặc ngữ cảnh cụ thể.

Trong luận văn này chúng ta sẽ tập trung vào việc truy hồi văn bản trong IR, nơi người dùng nhập một truy vấn văn bản và hệ thống trả về danh sách kết quả tìm kiếm được xếp hạng. Kết quả tìm kiếm có thể là các đoạn văn bản hoặc tài liệu toàn văn. Mục tiêu của hệ thống là xếp hạng kết quả tìm kiếm ưa thích của người dùng ở trên cùng. Vấn đề này là một vấn đề trọng tâm trong các tài liệu về IR, với những thách thức và giải pháp được hiểu rõ.

1.2. Quá trình truy hồi thông tin

Truy hồi thông tin là hoạt động thu thập, hiển thị thông tin liên quan dựa trên cácdữ liệu có sẵn hoặc từ thao tác cập nhật dữ liệu trên máy chủ [2].

Quá trình truy hồi thông tin là quá trình truy vấn dữ liệu từ nhu cầu của ngườidùng đến sự phản hồi của nguồn dữ liệu có sẵn được tạo ra từ trước.



Hình 1.1: Sơ đồ hiển thị quá trình truy hồi thông tin cơ bản [2]

Quá trình truy hồi ở sơ đồ trên chia làm 2 pha.

Pha 1 (Chỉ mục indexing): Từ bộ sưu tập tài liệu tạo ra các chỉ mục của tài liệu: Là quá trình thu thập tất cả các tài liệu liên quan "Document Collection" đến giaiđoạn lập chỉ mục "Indexer" kết thúc pha 1 là tạo danh mục các văn bản"Document Index".

Pha 2 (Truy hồi): Từ truy vấn của người sử dụng đến xử lý truy vấn và tìm tàiliệu phù hợp với truy vấn: Kết quả được xếp theo thứ tự mức độ liên quan.

Hệ truy hồi thông tin hoạt động theo phương thức sau

Giai đoạn đầu tiên là giai đoạn tiền xử lý, trong đó tài liệu thô của dữ liệu đượcxử lý thành các tài liệu được tách từ, phân đoạn (tokenized documents) và sau đó lập chỉ mục thành danh sách các vị trí của dữ liệu từ (postings per term).

Ở giai đoạn thứ hai, người dùng thực hiện một truy vấn (không có cấu trúc bằng ngôn ngữ tự nhiên) để mô tả nhu cầu tìm kiếm thông tin của mình. Hệ thống truy hồi bắt đầu truy vấn và so sánh để tìm các tài liệu và thông tin có liên quan đến truy vấn. Cácthủ tục được sử dụng để quyết định các phần tử thông tin có liên đến truy vấn dựa trên việc biểu diễn của truy vấn và các phần tử thông tin có chứa phần tử ngôn ngữ chỉ mục [25].

Trong giai đoạn cuối cùng, các tài liệu và thông tin tìm thấy được hiển thị trong một danh sách các tài liệu và được sắp xếp theo thứ tự phù hợp (ranked documents). Thông thường các tài liệu và yếu tố thông tin liên quan nhất được xếp trên các yếu tố ít liên quan hơn. Tùy thuộc vào hệ thống truy hồi thông tin khác nhau, chúng hiển thịthông tin theo những cách khác nhau. Ví dụ: có những hệ thống chỉ hiển thị tên tiêu đềvà đường dẫn đến tài liệu đó, hoặc có những hệ thống hiển thị cả tên và đường dẫn cùng một chút nội dung liên quan đến truy vấn, hoặc có những hệ thống dùng để lấy thông tin trên mạng liên kết đến các trang web khác nhau [2].

Nhiều hệ thống thông tin cũng bao gồm các cơ chế cho phép người dùng cung cấp phản hồi về chất lượng của kết quả trả về. Bằng cách sử dụng phản hồi, hệ thống sẽ cố gắng thích ứng và cố gắng tìm ra những kết quả tốt nhất cho truy vấn.

1.2.1. Giai đoạn tiền xử lý

- Tiền xử lý tài liệu là quy trình chuyển đổi văn bản. Quy trình này là một trong những bước quan trọng ảnh hưởng đến hiệu quả của hệ thống IR, nếu tiền xử lý không phù hợp có thể ảnh hưởng đến độ chính xác của phân loại văn bản. Ta tiến hành các bước sau:

  - O Phân tích từ vựng là quá trình thay đổi các ký tự trong tài liệu thành tập một tập các từ được chọn làm từ chỉ mục bằng cách loại bỏ các chữ số, dấu gạch nối, ký hiệu đặc biệt, dấu câu và chữ viết in hoa viết thường, chuẩn hóa các từ viết tắt [26].
  - O Loại bỏ từ dừng (stopword) làm giảm kích cỡ cấu trúc chỉ mục. Tiến hành loại bỏ các từ không ý nghĩa mà thường xuyên xuất hiện trong tài liệu.
  - O Lấy gốc từ là thu gọn một từ về dạng ngữ pháp gốc của nó. Ví dụ có nhiều từ sẽ mang ý nghĩa tương đồng ta chỉ cần xác định chọn một từ làm trọng tâm thể hiện nội dung chính.

- Đánh chỉ mục: cho phép tích hợp ngữ nghĩa thu được từ kho dữ liệu riêng.

Cấu trúc chỉ mục gồm tập hợp các thuật ngữ đã xử lý, cùng với danh sách tài liệuchứa chúng và trọng số của chúng. Trọng số của các thuật ngữ có thể là số lần xuất hiện của chúng trong một tài liệu. Tần suất xuất hiện càng lớn thì tầm quan trọng của chúng càng lớn.

1.2.2. Giai đoạn thu thập

- ***Xử lý truy vấn:*** Trong IR, một "yêu cầu" có thể được viết bằng ngôn ngữ tự nhiên, dưới dạng từ khóa hoặc dưới dạng toán tử Boolean. Bước đầu tiên trong giai đoạn truy hồi là xử lý truy vấn của người dùng cũng như xử lý trước các tài liệu văn bản gốc. Xử lý văn bản là thao tác chính để thể hiện nhu cầu của người dùng. Kết quả sẽ là một danh sách các từ.

- ***Tìm kiếm:*** Trong giai đoạn tìm kiếm, thuật ngữ thu được từ quá trình xửlý văn bản sẽ được sử dụng để xác định, thông qua chỉ mục và danh sách tài liệu thuật ngữ đó xuất hiện. Tùy thuộc vào từng loại truy vấn và tần suất xuất hiện của các từ đó trong tìm kiếm, một tập tài liệu sẽ được thu thập gồm tất cả các từ hoặc một số từ. Vì vậy, tìm kiếm là quá trình so khớp các thuật ngữ trong truy vấn. Kết quả so khớp phù hợp sẽ xếp hạng tài liệu theo thứ tự giảm dần về mức độ phù hợp để truyền tải nội dung đến người dùng.

- ***Xếp hạng:*** Các tài liệu thu thập được sẽ được xếp hạng tùy theo mức độ phù hợp với nội dung truy vấn. Việc đánh giá phụ thuộc chính xác vào thuật toán xếp hạng, thuật toán sẽ tính toán kết quả thực tế cho từng tài liệu liên quan. Tài liệu có trọng số xuất hiện càng lớn thì được xem là liên quan nhiều hơn. Và thứ tự kết quả trả về sẽ được sắp xếp theo trình tự giảm dần của giải thuật. Từ đó, người dùng có thể lựa chọn để xem xét tài liệu có nội dung liên quan nhất từ trên xuống. Vì vậy, giải thuật xếp hạng được xem là phần quan trọng trong IR.

- ***Phản hồi về độ liên quan:*** Quá trình truy hồi có thể lặp đi lặp lại, khi hệ thống nhận được phản hồi từ người dùng chẳng hạn như đánh giá mức độ phù hợp củatài liệu được xếp hạng cao. Từ đó, nó sẽ cải thiện tính đại diện của nhu cầu thông tin và đưa ra kết quả tốt hơn cho việc xếp hạng tài liệu.

1.3. Giới thiệu phần mềm Lucene

Hiện nay trên thế giới có một số thư viện mã nguồn mở chuyên hỗ trợ xây dựng hệ thống tìm kiếm thông tin như: Lucene, Egothor, Xapian, MG4J,… và Lucene chính làthư viện mã nguồn mở được nhiều cá nhân, tổ chức lựa chọn và sử dụng nhiều nhất [24].

Ví dụ: CNET dùng Lucene để tìm kiếm danh sách nhiều thể loại sản phẩm, Wikipedia thì dùng Lucene để tìm kiếm nội dung toàn văn bản.

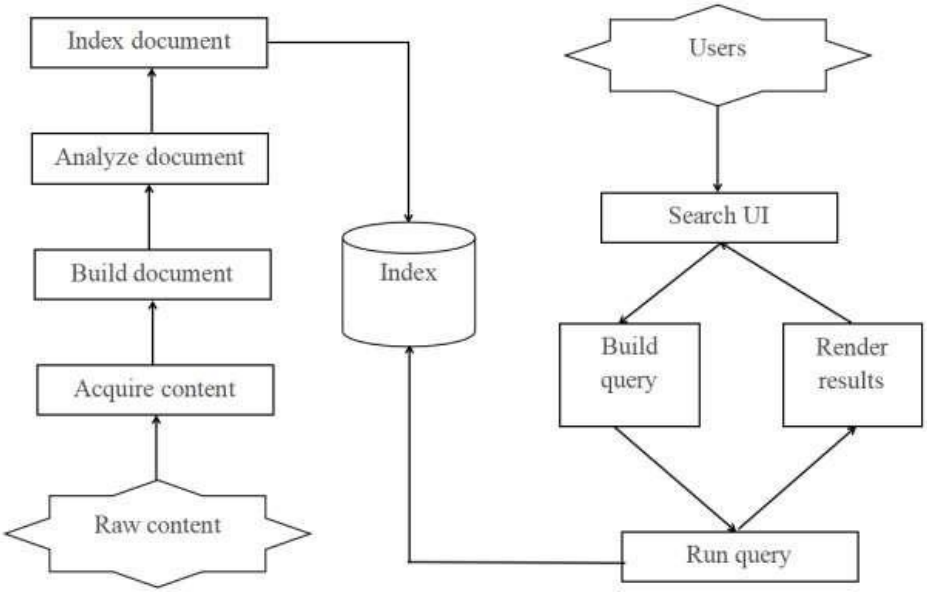
Bên cạnh đó, Elasticsearch và Solr là hai công cụ tìm kiếm khá phổ biến

đã được xây dựng và phát triển dựa trên nền tảng của Lucene. Do đó, tôi cũng đã sử dụng Lucenetrong đề tài xây dựng hệ thống truy hồi thông tin thử nghiệm để tìm kiếm thông tin truy hồi dữ liệu. Luận văn này kế thừa thư viện mã nguồn mở Lucene để xây dựng truy hồi với hai thành phần chính là lập chỉ mục và tìm kiếm văn bản. Tìm hiểu tính năng, hoạt động của mã nguồn mở Lucene và sử dụng Lucene.Net để xây dựng thử nghiệm hệ thống tìm kiếm truy hồi thông tin.

1.3.1 Tổng quát

Lucene là thư viện hoạt động khá hiệu quả trong hệ thống truy hồi thông tin. Lucene là một dự án mã nguồn mở hoàn chỉnh, có thể tải xuống miễn phí được cài đặt bằng Java; Lucene là thành viên thuộc các dự án của Apache Jakata, được đăng ký bản quyền từ Apache Software License [25].

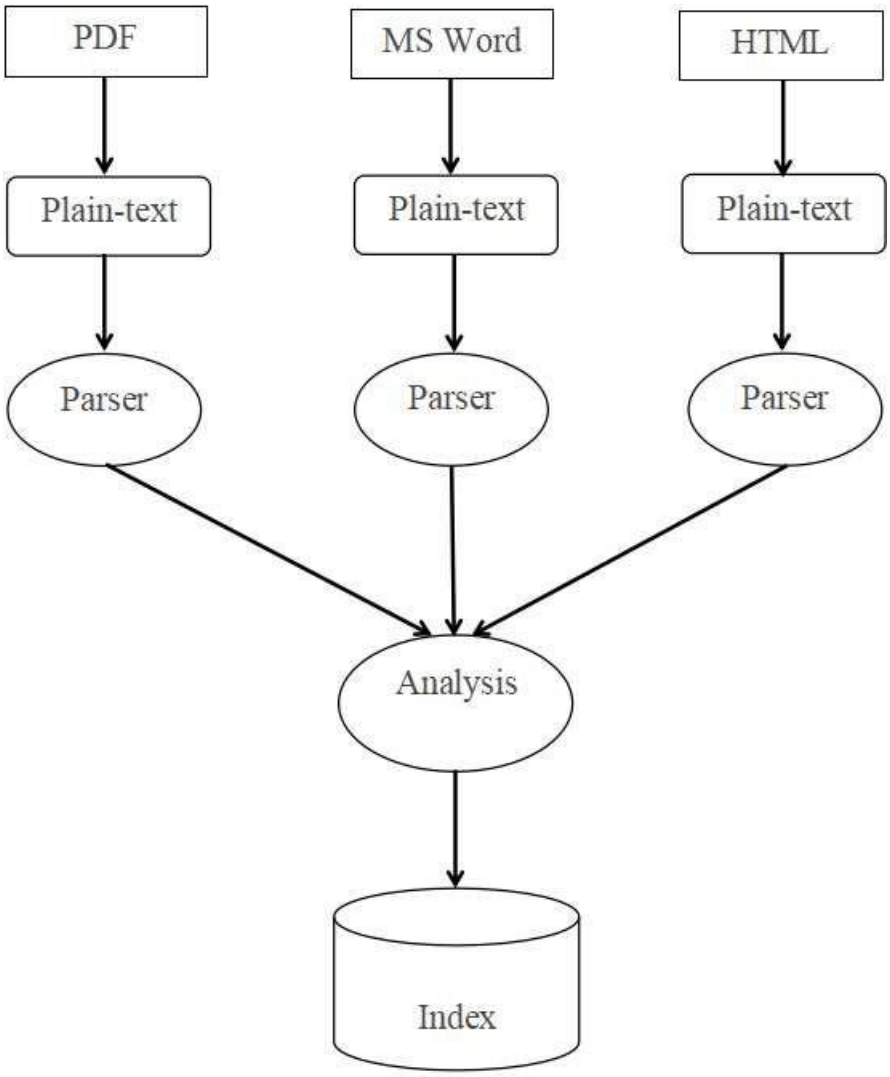
Lucene cho phép xử lý dữ liệu đầu vào dưới dạng văn bản nhằm tạo ra tập chỉ mục và cung cấp phương thức tìm kiếm trên tập chỉ mục đó. Đồng thời, nó cũng cho phép người dùng kế thừa, phát triển và thao tác với nhiều ngôn ngữ khác nhau.



Hình 1.2: Sơ đồ Lucene trong hệ thống tìm kiếm truy hồi thông tin [25]

1.3.2 Quy trình lập chỉ mục

Đầu tiên, chúng ta chuyển đổi tất cả nội dung trong các file dữ liệu như: PDF, MS Word, HTML, … sang nội dung chứa dữ liệu dạng văn bản (text). Sau đó Lucene tiến hành phân tích và xử lý dữ liệu, loại bỏ từ vô nghĩa, tách các từ, cụm từ,… sau khi dữ liệu được phân tích xong sẽ chuyển sang đánh chỉ mục (Index) [19].



Hình 1.3: Quy trình đánh chỉ mục [18]

1.3.3 Các toán tử đánh chỉ mục cơ bản

Lucene hỗ trợ các toán tử đánh chỉ mục cơ bản như sau:

- Thêm tài liệu mới: mỗi Document chứa nhiều Fields cùng tồn tại và mỗi Fields có nhiều giá trị khác nhau.
- Xóa tài liệu (Remove Document): sử dụng IndexReader với phương thức delete () sẽ dễ dàng xóa bỏ tài liệu được chọn ra khỏi chỉ mục.
- Cập nhật tài liệu: được thực hiện bằng việc xóa bỏ tài liệu sau đó là thêm mới tài liệu.

1.3.4 Tối ưu hóa đánh chỉ mục

Tối ưu hóa là quy trình trộn nhiều file chỉ mục với nhau nhằm giảm thiểu thời gian đọc chỉ mục trong suốt quá trình tìm kiếm. Sử dụng API của Lucene cụ thể là Optimize của đối tượng sử dụng Index Writerbta có thể dễ tối ưu nó. Tuy nhiên, điều này chỉ có thể làm tăng tốc độ tìm kiếm trên chỉ mục đã có mà không tác dụng tới tốc độ đánh chỉ mục.

1.3.5 Bộ phân tích Analyzer

Analyzer phân tích văn bản thành tokenizer, nó là quá trình trích xuất các từ, loại bỏ các dấu chấm câu, chuyển đổi tất cả từ trong văn bản thành chữ thường, chuẩn hóa chữ viết hoa (lower casing or normalizing), loại bỏ các từ dừng thông thường (stop word or common word), giảm số lượng từ văn bản đưa vào (root form or stremming). Toàn bộ quá trình này được gọi là tokenization, chuyển đổi văn bản thành các đoạn văn bản gọi là các token. Tokens được kết hợp với các tên file gọi là terms.

Sau quá trình tạo ra terms, terms sẽ là những khối dữ liệu được dùng để tìm kiếm trực tiếp. Chính vì vậy chọn bộ phân tích Analyzer đúng là vấn đề quan trọng của quá trình duy trì phần mềm tìm kiếm. Ngôn ngữ cũng là yếu tố chính được đề cập đến để chọn bộ phân tích vì chúng đều có đặc tính riêng và duy nhất trong từng ngôn ngữ.

1.4. Các phương pháp giải quyết vấn đề truy hồi thông tin

- Các phương pháp tiếp cận dựa trên thống kê, các tài liệu thu thập được xếp hạng cao vì những tài liệu được xác định là phù hợp nhất cho truy vấn.

- Các loại hướng tiếp cận là Mô hình truy hồi Boolean (Boolean Retrieval Model) và mô hình không gian Vec-tơ (Vector Space Model).

+ Truy hồi Boolean dựa trên mệnh đề logic.

+ Mô hình Vec-tơ không gian, là các tài liệu các truy vấn được biểu diễn dưới dạng Vec-tơ. Các từ khóa được lập chỉ mục và mối tương quan giữa tài liệu với truy vấn được tính bằng khoảng cách hình học giữa chúng.

1.5. Đánh giá hiệu quả của việc truy hồi thông tin

Việc đánh giá mức độ chính xác của kết quả gọi là đánh giá truy hồi thông tin. Cùng với thước đo hiệu suất phần mềm, hiệu suất truy hồi là vấn đề then chốt của hệ thống IR.

Đánh giá IR được thực hiện bằng cách truy vấn tập tham chiếu chuẩn hoá. Các tập tham chiếu này bao gồm tập dữ liệu, tập yêu cầu thông tin tham chiếu và tập dữ liệu liên quan tương ứng. Dữ liệu liên quan đến yêu cầu mẫu sẽ được xác định bởi các chuyên gia. Sự tương đồng giữa tập dữ liệu thu thập được và tập dữ liệu liên quan, được so sánh và định lượng theo các tiêu chí đánh

giá của tập kiểm tra và đó là chất lượng của chiến lược IR cần được xem xét.

Hình 1.4: Các tài liệu thu nhận được so với tài liệu liên quan

Nhận xét: Phần giao nhau giữa hai hình tròn nhỏ bên trong chính là phần mà dữ liệu tối ưu hóa được chọn.

Chương 2: CHỈ MỤC VĂN BẢN TỰ ĐỘNG

Trong chương này nhằm mục đích phân loại tự động các văn bản thành các danh mục xác định trước và sắp xếp chúng để cho việc truy hồi linh hoạt và hiệu quả hơn.

2.1 Học máy

Trong thời đại phát triển kỹ thuật số, để xử lý khối lượng dữ liệu lớn nhằm đáp ứng nhu cầu truy hồi thông tin, chúng ta cần phải nhờ đến ứng dụng của máy học dùng để phân tích dữ liệu dưới dạng văn bản.

Học máy là một ứng dụng của Trí tuệ nhân tạo, là một lĩnh vực giúp hệ thống tự động hiểu được dữ liệu từ dữ liệu được đào tạo mà chúng ta không cần lập trình cụ thể. Học máy chia làm 3 phần: học có giám sát, học bán giám sát và học không giám sát.

Học máy có giám sát là thuật toán được dùng để dự đoán tập dữ liệu đầu ra dựa vào tập dữ liệu đã được huấn luyện. Phương pháp phân loại và hồi quy là hai loại của máy học có giám sát. Phân loại là chia dữ liệu theo từng nhóm rồi đưa ra kết quả dự đoán, hồi quy thì cho ra kết quả dự đoán là một số thực cụ thể.

Học máy không giám sát là thuật toán dự đoán dữ liệu đầu ra dựa vào duy nhất tập dữ liệu đầu vào, dữ liệu đầu vào sẽ không được dán nhãn hoặc kết quả đầu ra. Máy học không giám sát bao gồm phân nhóm và tích hợp. Thuật toán phân nhóm là phân tập dữ liệu thành các nhóm nhỏ dựa vào các liên quan của dữ liệu trong nhóm. Thuật toán tích hợp sẽ tìm ra một số quy luật trên tập dữ liệu để tiến hành khai phá dữ liệu.

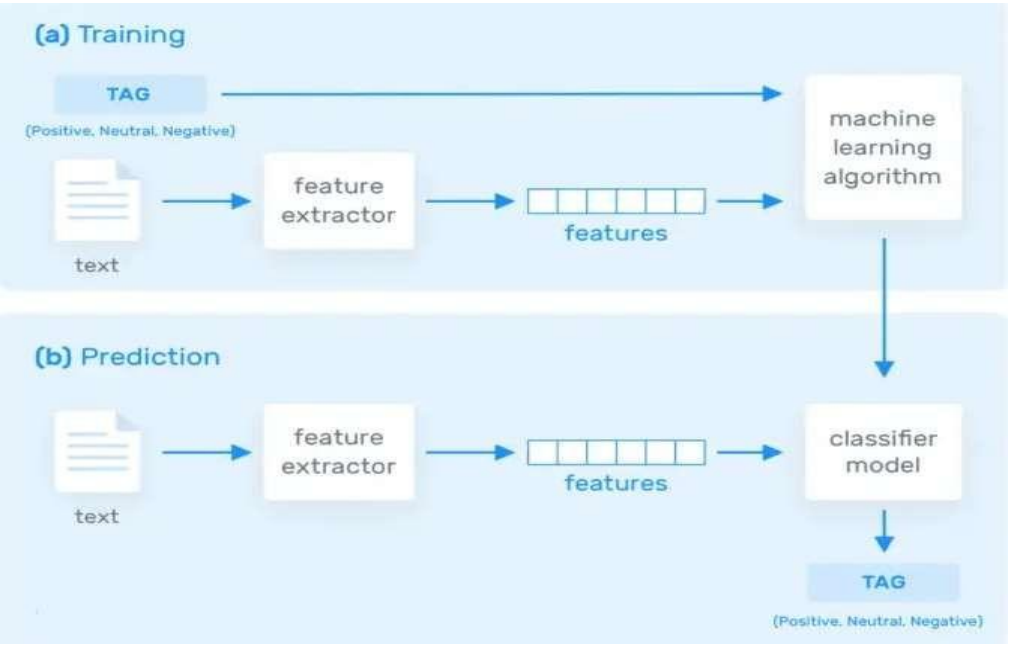
Học máy bán giám sát là thuật toán kết hợp của cả hai thuật toán có giám sát và không giám sát. Dữ liệu chia một phần được gán nhãn, phần còn lại thì không được gán nhãn.

Trong nghiên cứu này, tôi chọn phương pháp học máy có giám sát để áp dụng phân tích nội dung văn bản và trả về kết quả có nội dung liên quan đến truy vấn.

2.2 Phân loại văn bản

Phân loại văn bản được áp dụng trong một số miền như: lập chỉ mục tài liệu dựa trên vốn từ vựng được kiểm soát, lọc tài liệu, phân loại cảm giác tài liệu...

Cách tiếp cận chủ đạo để phân loại văn bản dựa vào Kỹ thuật học máy: là một quy trình quy nạp chung tự tạo bộ phân loại bằng cách học từ một tập hợp các tài liệu đã được phân loại trước dựa vào các đặc điểm của danh mục.



Hình 2.1: Mô hình phân loại tiếng Việt tự động với Machine learning [22]

- Giai đoạn (a): Huấn luyện (training) là giai đoạn học tập của mô hình phân loại văn bản. Ở bước này, mô hình sẽ học từ dữ liệu có nhãn (trong hình trên nhãn là Positive, Negative, Neutral). Dữ liệu văn bản sẽ được số hóa thông qua bộ trích xuất đặc trưng (feature extractor) để mỗi mẫu dữ liệu trong tập huấn luyện trở thành 1 Vec-tơ nhiều chiều (đặc trưng). Thuật toán máy học sẽ học

và tối ưu các tham số để đạt được kết quả tốt trên tập dữ liệu này. Nhãn của dữ liệu được dùng để đánh giá việc mô hình học tốt không và dựa vào đó để tối ưu.

- Giai đoạn (b): Dự đoán (prediction), là giai đoạn sử dụng mô hình học máy sau khi nó đã học xong. Ở giai đoạn này, dữ liệu cần dự đoán cũng vẫn thực hiện các bước trích xuất đặc trưng. Mô hình đã học sau đó nhận đầu vào là đặc trưng đó và đưa ra kết quả dự đoán.

2.2.1 Xử lý ngôn ngữ tự nhiên – thuật toán tách từ (tokenizer)

Xử lý ngôn ngữ tự nhiên (NLP) là một phần của Trí tuệ nhân tạo (AI) cung cấp cho máy tính khả năng hiểu ngôn ngữ viết và nói của con người. Ví dụ như các ứng dụng của NLP trong kiểm tra chính tả, tự động điền từ, phát hiện thư rác, ... Tuy nhiên, đó là những hoạt động của máy vi tính với các con số chứ không phải các chữ cái các từ hay các câu. Vì vậy, để làm việc với một lượng lớn dữ liệu văn bản có sẵn, tiền xử lý văn bản (text pre-processing) là quá trình cần thiết giúp làm sạch văn bản. Bản thân tiền xử lý văn bản bao gồm nhiều giai đoạn, và một trong số đó là tách từ (hay còn gọi là Tokenization).

Tokenization (tách từ) là một bước quan trọng trong quá trình tiền xử lý văn bản. Cho dù bạn đang làm việc với các kỹ thuật NLP truyền thống hay sử dụng các kỹ thuật học sâu nâng cao thì vẫn không thể bỏ qua bước này. Hiểu đơn giản, tokenization là quá trình tách một cụm từ, câu, đoạn văn, một hoặc nhiều tài liệu văn bản thành các đơn vị nhỏ hơn. Mỗi đơn vị nhỏ hơn này được gọi là Tokens.

Có thể xem tokens là các khối xây dựng của NLP và tất cả các mô hình NLP đều xử lý văn bản thô ở cấp độ các Tokens. Chúng được sử dụng để tạo từ vựng trong một kho ngữ liệu (một tập dữ liệu trong NLP). Từ vựng này sau đó được chuyển thành số (ID) và giúp chúng ta lập mô hình. Tokens có thể là bất cứ thứ gì là một từ (word), một từ phụ (sub-word) hoặc thậm chí là một ký tự

(character). Các thuật toán khác nhau tuân theo các quy trình khác nhau trong việc thực hiện mã hóa và sự khác biệt giữa ba loại tokens này sẽ được chỉ ra dưới đây.

Ví dụ: Câu gốc là "Tôi là người Việt Nam."

Phân loại các kỹ thuật tách từ dựa trên ví dụ cụ thể

Thuật toán mã hóa dựa trên từ (word-based tokenization algorithm) chia câu thành các từ: ["Tôi", "là", "người", "Việt Nam."]

Thuật toán mã hóa dựa trên từ phụ (subword-based tokenization algorithm) chia câu thành các từ khóa phụ: ["Tôi", "là", "người", "Việt", " Nam."]

Thuật toán mã hóa dựa trên ký tự (character-based tokenization algorithm) chia câu thành các ký tự, ở đây là từng chữ cái một.

["T", "ô", "i", "l", "a", "n", "g", "ư", "ơ", "i", "V", "i", "ê", "t", "N", "a", "m."]

Ba kỹ thuật mã hóa này hoạt động khác nhau và có những ưu điểm và nhược điểm riêng sẽ được phân tích cụ thể như sau:

Word-based tokenization

Là kĩ thuật tokenization được sử dụng phổ biến trong phân tích văn bản. Nó chia một đoạn văn bản thành các từ (ví dụ tiếng Anh) hoặc âm tiết (ví dụ tiếng Việt) dựa trên dấu phân cách. Dấu phân cách hay được dùng chính là dấu cách trắng. Tuy nhiên, cũng có thể tách văn bản không theo dấu phân cách. Giả sử tách từ trong tiếng Việt vì một từ trong tiếng Việt có thể chứa hai hoặc ba âm tiết được ghép nhau bởi dấu cách.

Việc tách từ có thể thực hiện dễ dàng bằng cách sử dụng phương thức split () của RegEx hoặc Python. Ngoài ra, có rất nhiều thư viện Python – NLTK, spaCy, Keras, Gensim, có thể giúp bạn thực hiện việc này một cách thuận tiện.

Thực tế, các mô hình NLP sử dụng các phương pháp tách từ phù hợp theo từng ngôn ngữ. Tuỳ thuộc vào từng bài toán, mà cùng một văn bản có thể được xử lý dưới các loại tokens khác nhau. Mỗi token thường có tính duy nhất và được biểu diễn bằng một ID, các ID này là một cách mã hoá hay cách định danh token trên không gian số.

Hạn chế của kỹ thuật này là nó dẫn đến một kho ngữ liệu khổng lồ và một lượng từ vựng lớn, khiến mô hình cồng kềnh hơn và đòi hỏi nhiều tài nguyên tính toán hơn. Bên cạnh đó, một hạn chế nữa là liên quan đến các từ sai chính tả. Nếu kho ngữ liệu có từ "knowledge" viết sai chính tả thành "knowldge", mô hình sẽ gán token OOV cho từ sau đó. Do đó, để giải quyết tất cả những vấn đề này, các nhà nghiên cứu đã đưa ra kỹ thuật mã hóa dựa trên ký tự.

Character-based tokenization

Mã hóa dựa trên ký tự chia văn bản thô thành các ký tự riêng lẻ. Logic đằng sau mã hóa này là một ngôn ngữ có nhiều từ khác nhau nhưng có một số ký tự cố định. Điều này dẫn đến một lượng từ vựng rất nhỏ. Ví dụ tiếng Anh có 256 ký tự khác nhau (chữ cái, số, ký tự đặc biệt) trong khi chứa gần 170.000 từ trong vốn từ vựng. Do đó, mã hóa dựa trên ký tự sẽ sử dụng ít token hơn so với mã hóa dựa trên từ.

Một trong những lợi thế chính của mã hóa dựa trên ký tự là sẽ không có hoặc rất ít từ không xác định hoặc OOV. Do đó, nó có thể biểu diễn các từ chưa biết (những từ không được nhìn thấy trong quá trình huấn luyện) bằng cách biểu diễn cho mỗi ký tự. Một ưu điểm khác là các từ sai chính tả có thể được viết đúng chính tả lại, thay vì có thể đánh dấu chúng là mã thông báo OOV và làm mất thông tin.

Loại mã hóa này khá đơn giản và có thể làm giảm độ phức tạp của bộ nhớ và thời gian. Vì vậy, liệu nó có phải thuật toán tốt nhất hay hoàn hảo để tách từ?

Câu trả lời là không (ít nhất là đối với Ngôn ngữ tiếng Anh)! Một ký tự thường không mang bất kỳ ý nghĩa hoặc thông tin nào như một từ. Ngoài ra, tuy kỹ thuật này giúp giảm kích thước từ vựng nhưng lại làm tăng độ dài chuỗi trong mã hóa dựa trên ký tự. Mỗi từ được chia thành từng ký tự và do đó, chuỗi mã hóa dài hơn nhiều so với văn bản thô ban đầu. Vì vậy, có thể thấy, dù đã giải quyết được rất nhiều thách thức mà mã hóa dựa trên từ gặp phải, mã hóa dựa trên ký tự vẫn có một số vấn đề nhất định.

Subword-based tokenization

Mã hóa dựa trên từ khóa phụ, là một giải pháp nằm giữa mã hóa dựa trên từ và kýtự. Ý tưởng chính là giải quyết đồng thời các vấn đề của mã hóa dựa trên từ (kích thước từ vựng rất lớn, có nhiều tokens OOV, sự khác biệt trong ý nghĩa của các từ rất giống nhau) và mã hóa dựa trên ký tự (chuỗi rất dài và token riêng lẻ ít ý nghĩa hơn).

Các thuật toán mã hóa dựa trên từ khóa phụ sử dụng các nguyên tắc sau:

- Không chia các từ thường dùng thành các từ phụ nhỏ hơn.
- Chia các từ hiếm thành các từ phụ có ý nghĩa.

Hầu hết các mô hình tiếng Anh đều sử dụng các dạng thuật toán của mã hóa từ phụ, trong đó, phổ biến là WordPeces được sử dụng bởi BERT và DistilBERT, Unigramcủa XLNet và ALBERT, và Bye-Pair Encoding của GPT-2 và RoBERTa.

Mã hóa dựa trên từ khóa phụ cho phép mô hình có kích thước từ vựng phù hợp vàcũng có thể học các biểu diễn độc lập theo ngữ cảnh có ý nghĩa. Mô hình thậm chí có thể xử lý một từ mà nó chưa từng thấy trước đây vì sự phân tách có thể dẫn đến các từ phụ đã biết.

Như vậy, trên đây là cách các phương pháp mã hóa phát triển theo thời

gian để đáp ứng nhu cầu ngày càng tăng của NLP và đưa ra các giải pháp tốt hơn cho các vấn đề.

Trong luận văn này, chúng ta sẽ đề cập đến vấn đề tách từ bằng ngôn ngữ tiếng Việt và tiếng Anh vì dữ liệu chúng ta tìm kiếm truy hồi sẽ là tài liệu bằng ngôn ngữ tiếng Việt và tiếng Anh.

Bài toán tách từ là một trong các bài toán cơ bản đầu tiên dùng trong việc xử lý ngôn ngữ sau:

· *Phân tích hình thái học (morphological analysis)*

o Phân tích các dấu

o Nhận dạng tên

o Xác định ranh giới ngôn ngữ

· *Phân tích ngữ pháp (PARSER)*

o Gán nhãn loại từ

o Gán nhãn ranh giới ngôn ngữ

o Gán nhãn các quan hệ cú pháp

· *Xử lý văn bản*

o Kiểm tra chính tả

o Kiểm tra lỗi ngữ pháp

o Phân loại văn bản

o Tóm tắt văn bản

o Hiểu văn bản

o Khai thác văn bản

- *Nguồn lực hỗ trợ:*

· Từ điển tiếng Việt, tiếng Anh

· Kho ngữ liệu tiếng Việt, tiếng Anh được tách từ để hỗ trợ quá trình đào tạo.

2.2.2 Loại bỏ từ dừng

Từ dừng là những từ có tần suất rất cao được xem là không hữu ích cho việc tìm kiếm. Chúng có rất ít trọng số ngữ nghĩa. Tất cả được lập trong một danh sách được gọi là danh sách từ dừng. Ví dụ các từ dừng như: là, của, trong, trên, tại, ở,… Theo định luật Zipf, một danh sách có vài chục từ dừng sẽ làm giảm kích thước của chỉ mục đảo ngược gần một nửa. Tuy nhiên, việc loại bỏ từ dừng có thể gây ảnh hưởng đến ý nghĩa của việc tìm kiếm các cụm từ. Ví dụ, nếu ta loại bỏ chữ "B" khỏi cụm "Vitamin B" thì từ còn lại không còn ý nghĩa nữa.

2.3 Chỉ mục văn bản

2.3.1 Tổng quan

Nhằm cải thiện hiệu quả thời gian truy hồi thông tin và giảm không gian lưu trữ trong hệ thống ta cần phải lập chỉ mục cho thư viện tài liệu, hay nói cách khác là rút trích văn bản. Tuy nhiên, khi ta thực hiện thay đổi nội dung hoặc thêm tài liệu mới ta cũng cần cập nhật lại tài liệu chỉ mục. Để thực hiện với nguồn dữ liệu lớn chúng ta sẽ dùng phương pháp lập chỉ mục văn bản tự động.

Như vậy, lập chỉ mục là quá trình phân tích, rút trích từ, cụm từ và thuật ngữ thích hợp có khả năng đại diện cho nội dung của tài liệu. Sau khi thực hiện rút trích thì nội dung đó được lưu trữ vào danh sách chỉ mục, khi thực hiện tìm kiếm ta chỉ cần so khớp nội dung truy vấn với danh sách chỉ mục sẽ cho ra kết quả nhanh chóng và chính xác hơn việc ta so sánh với tất cả các từ không có nghĩa trong văn bản [3].

Các bước cơ bản trong xây dựng chỉ mục:

→ Thu thập tài liệu cần chỉ mục

→ Mã hóa văn bản

→ Thực hiện tiền xử lý ngôn ngữ của mã thông báo

→ Lập chỉ mục các tài liệu thuật ngữ xuất hiện.

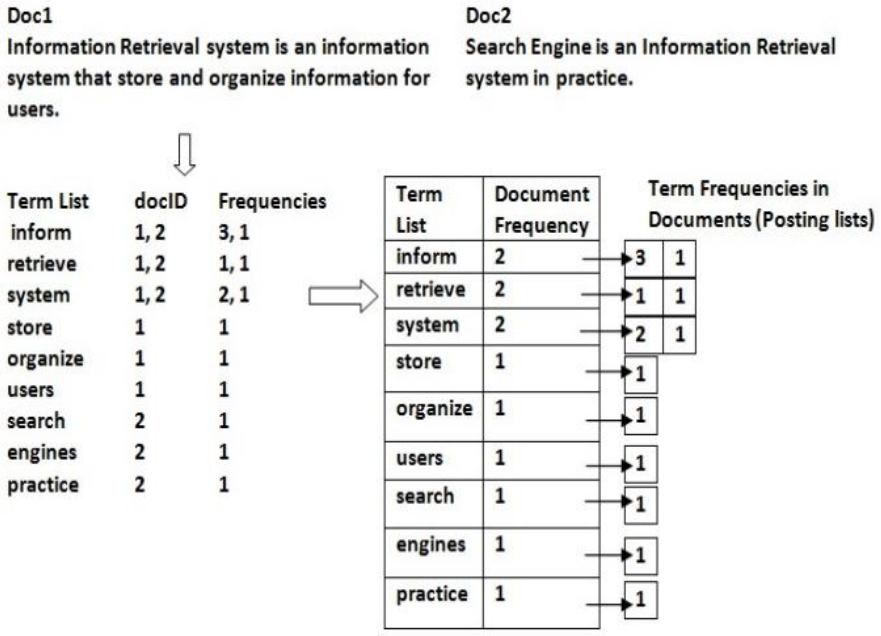
Bên cạnh đó ta cần xác định các công việc cụ thể khi thực hiện chỉ mục như sau:

+ Xác định từ, cụm từ, thuật ngữ có khả năng đại diện cho nội dung tài liệu.

+ Đánh trọng số cho từ hoặc cụm từ này, trọng số sẽ phản ảnh tầm quan trọng của từ trong một tài liệu.

Các ví dụ về loại chỉ mục văn bản

Ví dụ 1: Xây dựng chỉ mục cho hai đoạn văn bản Doc1 và Doc2 bằng cách sắp xếp và nhóm. Các chuỗi từ, thuật ngữ được sắp xếp theo thứ tự chữ cái (bên cột trái) và gắn thẻ ID cho từng từ. Các trường hợp của cùng một thuật ngữ sau đó được nhóm theo từ và mã ID. Các thuật ngữ và mã ID được tách ra (bên phải). Danh sách từ điển các thuật ngữ và mũi tên chỉ đến danh sách bài đăng cho từng thuật ngữ.

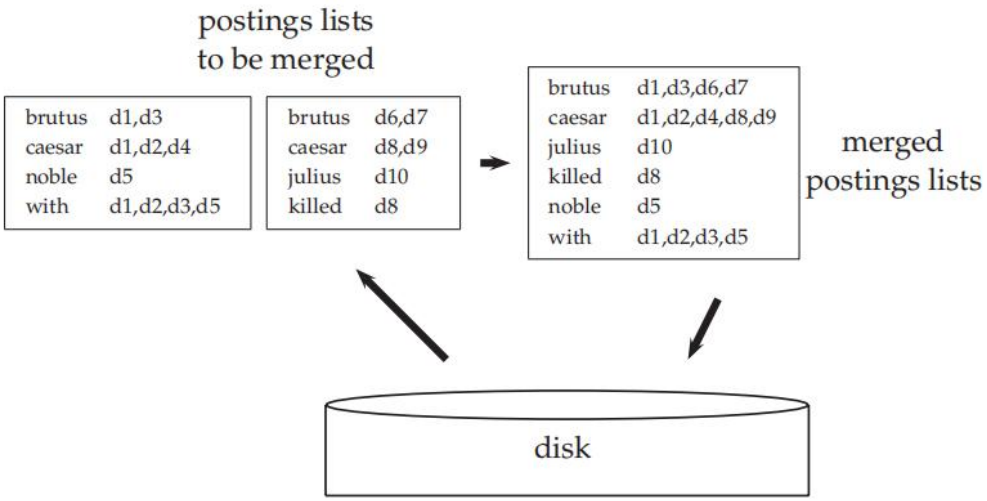


Hình 2.2: Xây dựng chỉ mục bằng cách sắp xếp và nhóm [5]

Như vậy, chỉ mục văn bản là danh sách lưu trữ thông tin tóm tắt. Tại đây sẽ nêu rõ tầng suất tài liệu xuất hiện của mỗi kỳ hạn. Luận văn sử dụng chỉ mục văn

bản nhằm cải thiện hiệu quả thời gian truy vấn sau này, và để trọng số trong các mô hình truy hồi được xếp hạng. Mỗi danh sách đăng tin lưu trữ danh sách các tài liệu trong đó có một thuật ngữ xuất hiện và có thể lưu trữ thông tin khác, chẳng hạn như tầng suất thuật ngữ (tầng suất xuất hiện của mỗi thuật ngữ trong mỗi tài liệu) hoặc vị trí của thuật ngữ trong mỗi tài liệu.

Ví dụ 2: Lập chỉ mục dựa trên sắp xếp bị chặn. Là việc thực hiện chỉ mục với hai lượt tiếp cận, ta gộp từ vựng trong lượt đầu tiên và xây dựng chỉ mục đảo ngược trong lượt thứ hai. Các thuật toán chỉ mục thực hiện trong một lần truyền dữ liệu.



Hình 2.3: Mô hình hợp nhất trong chỉ mục dựa trên sắp xếp và bị chặn [3]

Hợp nhất trong lập chỉ mục dựa trên sắp xếp bị chặn. Hai khối danh sách bài đăng sẽ được hợp nhất" được tải từ đĩa vào bộ nhớ, được hợp nhất trong bộ nhớ và được ghi trở lại đĩa. Việc sử dụng hiển thị các điều khoản sẽ dễ đọc hơn thay vì dùng mã ID như ở chỉ mục theo cách sắp xếp và nhóm trong ví dụ 1.

2.3.2 Xác định từ, cụm từ quan trọng để lập chỉ mục

Từ, cụm từ quan trọng là từ có khả năng đại diện cho nội dung của tài liệu hay còn gọi mục từ. Mục từ là đơn vị cơ sở cho quá trình lập chỉ mục.

Việc lập chỉ mục tự bắt đầu với việc khảo sát tần số suất hiện của từ trong văn bản. Nếu tần số xuất hiện của từ đều bằng nhau thì ta không thể xác định các mục từ theo định lượng. Tuy nhiên trong văn bản các thì các mục từ thường xuất hiện bất thường nên ta dễ dàng xác định được tần số có mặt của chúng trong tài liệu. Theo định luật Zipf:

Định luật Zipf là mô hình thường được sử dụng để phân phối các thuật ngữ trong một tập hợp. Khi hiểu cách các từ được phân phối trên các tài liệu, giúp chúng ta mô tả đặc tính của các thuật toán nén danh sách đăng tin [3].

Ta có: Nếu t_1 là thuật ngữ phổ biến nhất trong tập hợp thì t_2 là thuật ngữ phổ biến tiếp theo,… thì tần số thu nhập cf của số hạng phổ biến thứ i tỷ lệ với 1/i:

$$cf_i \, \alpha \, \frac{1}{i}$$

Vì vậy, nếu số hạng thường xuyên nhất xuất hiện cf1 lần, thì số hạng thường xuyên thứ hai có số lần xuất hiện bằng một nữa, số hạng thường xuyên nhất thứ ba có số lần xuất hiện nhiều nhất,.. Trực giác là tầng số giảm rất nhanh theo cấp bậc. Công thức là một trong những cách đơn giản nhất để hình thức hóa sự giảm nhanh như vậy và nó là một mô hình hợp lý.

Tương tự, chúng ta có thể viết định luật Zipf như sau:

Sự xuất hiện của từ vựng có thể được định bởi hằng số “Thứ hạng_tần số” (Rank_Frequency).

Tần số xuất hiện * thứ hạng = Hằng số

Biểu thức định luật chỉ ra những hệ số có ý nghĩa của từ dựa vào đặc trưng của tần số xuất hiện của mục từ riêng rẻ trong văn bản tài liệu.

Một số đề xuất đính kèm luật Zipf

1. Cho tập hợp n tài liệu, trong mỗi tài liệu tính toán số lần xuất hiện các mục từ trong tài liệu đó.

Kí hiệu F_{ik} (Frequency): tần số xuất hiện của mục từ k trong tài liệu i.

2. Xác định tổng số tần số xuất hiện TF_k (Total Frequency) cho mỗi từ bằng cách cộng những tần số của mỗi mục từ duy nhất trên tất cả n tài liệu.

$$TF_k = \sum_{i}^{n} F_{ik}$$

3. Sắp xếp thứ tự giảm dần theo tập tần số xuất hiện. Quyết định giá trị ngưỡng cao và loại bỏ tất cả những từ có tập tần số xuất hiện cao trên ngưỡng này. Những từ bị loại bỏ là những từ xuất hiện phổ biến ở hầu hết các tài liệu. Chính là các từ dừng (Stop-Word).

4. Tương tự, loại trừ những từ được xem là có tần số xuất hiện thấp. Nghĩa là, xác định ngưỡng thấp và loại bỏ tất cả các từ có tần số nhỏ hơn giá trị này. Điều này sẽ loại bỏ các từ ít xuất hiện trong tập tài liệu, nên sự có mặt của các từ này cũng không ảnh hưởng đến việc thực hiện truy vấn.

5. Những từ xuất hiện trung bình còn lại được dùng ấn định tới những tài liệu như những mục từ chỉ mục.

2.3.3 Lập chỉ mục với Lucene

Lucene sẽ quản lý chỉ mục trên thư viện tài liệu động, nó sẽ cập nhật rất nhanh khi thêm hoặc xóa bỏ tài liệu ra khỏi thư viện. Lucene lập chỉ mục theo từ hoặc cụm từ, khi thực hiện tìm kiếm thì Lucene cũng sẽ tìm kiếm theo từ hoặc theo cụm từ tùy vào nội dung truy vấn.

Một từ, cụm từ hay còn gọi là thuật ngữ sẽ kết hợp tên trường với một mã thông báo. Các thuật ngữ bao gồm tên trường chứ không là nội dung văn bản trong thư viện tài liệu, nó có mã thông báo riêng.

Chỉ mục Lucene cung cấp ánh xạ từ điều khoản đến thư viện tài liệu. Chỉ mục đảo ngược là nó sẽ đảo ngược ánh xạ thông thường của tài liệu có chứa thuật ngữ. Cơ chế chỉ mục đảo ngược xác định điểm kết quả tìm kiếm, nếu tất cả các

thuật ngữ tìm kiếm đều ánh xạ đến cùng một tài liệu thì tài liệu đó có khả năng sẽ liên quan.

Trường chỉ mục

Về lý thuyết thì Lucene cung cấp tính năng tìm kiếm và lập chỉ mục nội dung tài liệu nhưng thực tế thì lập chỉ mục và tìm kiếm thực hiện thông qua các trường. Nội dung tài liệu là tập hợp của các trường. Một trường sẽ có ba phần chính là: tên trường, kiểu và giá trị.

Ví dụ: tài liệu về Mạch điện sẽ được thể hiện dưới dạng các trường sau:

+ Một trường là tên của tài liệu

+ Một trường là văn bản tóm tài liệu

+ Một trường là danh sách các từ khóa rút ra từ chuyên ngành điện - điện tử.

Tại một thời điểm tìm kiếm thì mỗi trường đều có tên gọi khác nhau.

Lập chỉ mục

Bước đầu tiên là xây dựng nội dung các trường đã được lập chỉ mục, sau đó thêm tài liệu vào danh sách chỉ mục. Các lớp liên quan đến chỉ mục sẽ chịu trách nhiệm thêm tài liệu vào chỉ mục và lưu trữ chỉ mục đó. Thư mục cung cấp có giao diện giống như một hệ thống tệp trong hệ điều hành. Một thư viện chỉ mục chứa bất kỳ chỉ mục con nào sẽ được gọi là phân đoạn. Và việc duy trì chỉ mục dưới dạng một tập hợp các phân đoạn cho phép Lucene thực hiện cập nhật và xóa tài liệu một cách dễ dàng, nhanh chóng.

2.4 Đánh trọng số

Tiêu chí tầm quan trọng của thuật ngữ được sử dụng để xếp hạng mức độ liên quan của tài liệu với truy vấn có chứa thuật ngữ đó. Một tài liệu có chứa nhiều thuật ngữ của một truy vấn có thể liên quan đến truy vấn đó.

Truy vấn văn bản tự do (free text query) là truy vấn mà trong đó các thuật ngữ được nhập tự do vào giao diện tìm kiếm mà không cần bất kỳ toán tử tìm kiếm nào. Loạitruy vấn này đơn giản và phổ biến trên web. Khi đó, sẽ có cơ chế tính toán mức độ phù hợp tương tự. Mức độ tương tự được tính toán dựa trên tổng tất cả các thuật ngữ của truyvấn và mức độ liên quan của từng thuật ngữ với tài liệu. Sự phù hợp đó được thể hiện thông qua trọng số tf và idf.

TF-IDF: Term Frequency – Inverse Document Frequency (Tần suất xuất hiện củatừ - Nghịch đảo tần suất của văn bản) là một kỹ thuật được sử dụng trong khai thác dữ liệu văn bản. Trọng số này sử dụng để đánh giá mức độ quan trọng của một từ trong mộtvăn bản. Giá trị cao thể hiện mức độ quan trọng cao và nó phụ thuộc vào số lần từ xuất hiện trong văn bản nhưng bù lại bởi tần suất xuất hiện của từ đó trong tập dữ liệu [2].

TF: Term Frequency (Tần suất xuất hiện của thuật ngữ) là số lần từ xuất hiện trong văn bản. Vì các văn bản có thể có độ dài khác nhau nên một số từ có thể xuất hiện nhiều lần hơn trong một văn bản dài hơn là văn bản ngắn. Do đó, term frequency thườngđược chia cho độ dài văn bản (tổng số từ trong một văn bản) [2].

$$tf(t,d) = \frac{f(t,d)}{\max\{f(w,d) : w \in d\}}$$

Trong đó:

tf(t, d): tần suất xuất hiện của từ t trong văn bản d

f(t,d): số lần xuất hiện của từ t trong văn bản d

$\max\{f(w,d): w \in d\}$: số lần xuất hiện của từ có nhiều lần xuất hiện nhất trong văn bản d

IDF: Inverse Document Frequency (Tần suất nghịch đảo văn bản) giúp đánh giá tầm quan trọng của một từ. Khi tính toán TF, tất cả các từ được coi là có tầm quan trọng như nhau. Nhưng có một số từ như "is", "of" và "that" thường xuất hiện nhiều lần nhưng mức độ quan trọng không cao. Vì vậy chúng ta cần giảm bớt tầm quan trọng của những từ này.

$$idf(t, D) = log \frac{|D|}{|\{d \in D: t \in d\}|}$$

Trong đó:

idf(t, D): giá trị idf của từ t trong tập văn bản D

|D|: Tổng số văn bản trong tập hợp D

|{d ∈ D : t ∈ d}|: thể hiện số văn bản trong tập D có chứa từ t.

Cụ thể, ta có công thức tính TF-IDF đầy đủ như sau:

TF-IDF(t, d, D) = tf(t, d) x idf(t, D)

Thì những từ có giá trị TF- IDF cao là những từ xuất hiện nhiều trong văn bản và xuất hiện ít trong văn bản khác. Điều này giúp lọc ra các từ phổ biến và giữ các từ có giá trị cao (từ khoá của văn bản đó).

2.5 Các mô hình xếp hạng truyền thống

Mô hình xếp hạng là một hệ thống giải quyết hoặc xây dựng các vấn đề IR khác nhau.

Thông thường, mô hình xếp hạng được viết vắn tắt bằng bốn chữ cái D, Q, F, R. Trong đó các chữ cái được định nghĩa như sau:

- **D** (Document collection) là một tập hợp các tài liệu. Mỗi tài liệu được mô hình hóa như một nhóm các điều khoản chỉ mục trong đó các điều khoản chỉ mục được định là độc lập với nhau.
- **Q** (Query collection) là một tập hợp các truy vấn. Các truy vấn do người sử dụng kích hoạt thuộc về tập hợp này. Nó cũng được mô hình hóa như một tập các thuật ngữ chỉ mục trong các trường hợp.
- **F** *(*Framework*)* trong mô hình mô tả tài liệu, giữa các câu truy vấn và mối quan hệ của chúng.
- **R** (Ranking function) là một hàm xếp hạng liên kết điểm (số thực) với một cặp (q_i, d_j) trong đó $q_i \in Q$ và $d_j \in D$. Cho truy vấn (q_i) các tài liệu sẽ được xếp hạng theo điểm.

2.5.1. Mô hình Boolean

Là mô hình truy hồi thông tin đơn giản. Mô hình dùng lý thuyết tập hợp và các toán tử Boolean như NOT, OR và AND. Các tài liệu được truy hồi là các tài liệu hoàn toàn phù hợp với truy vấn đã cho, tuy nhiên các tài liệu liên quan sẽ không được đề cập đến [3].

Hệ thống có 4 bản ghi như sau:

Ví dụ về Mô hình Boolean. Ta có 4 doc sau:

- Doc1 : Nhà ở Tây Ninh

- Doc2 : 40m^2
- Doc3 : 4 phòng ngủ
- Doc4 : Nhà ở Tây Ninh, 70m^2 Để trả lời cho câu query Nhà ở Tây Ninh, không phải 40m^2. Hệ thống sẽ phân tích thành Nhà AND Tây Ninh AND NOT 40m^2.

Dưới đây là phần trình bày có thể xuất hiện hoặc không xuất hiện trong các tài liệu doc của các index.

	Doc1	*Doc2*	*Doc3*	*Doc4*
Nhà	*1*	*0*	*0*	*1*
Tây Ninh	*1*	*0*	*0*	*1*
40m^2	*0*	*1*	*0*	*0*
Phòng ngủ	*0*	*0*	*1*	*0*
70m^2	*0*	*0*	*0*	*1*

Hình 2.4: Biểu diễn ví dụ trong Mô hình Boolean

Sử dụng phép toán AND ta có:

1001(Nhà)AND1001(Tây Ninh)AND1011(không 40m^2) = 1001

Kết quả hệ thống sẽ đưa ra cho bạn Doc4, Doc1.

Ưu điểm của Boolean:

- Đơn giản, dễ thao tác, dễ làm.
- Kết quả tìm kiếm nhanh, chính xác theo yêu cầu. Khối lượng dữ liệu tương

đối không quá lớn.

Nhược điểm:

- Không thao tác được những câu truy vấn phức tạp.
- Tốc độ tìm kiếm chậm với khối dữ liệu lớn.
- Tài liệu liên quan, từ đồng nghĩa không được hiển thị.
- Gặp khó khăn trong xếp hạng các bản ghi được truy vấn.

⇨ Như vậy Thuật toán Boolean được sử dụng phổ biến trong các quy mô tìm kiếm nhỏ như ổ cứng hay mail.

2.5.2 Mô hình không gian Vec-tơ

Để khắc phục nhược điểm mô hình Boolean gặp khó khăn trong việc xếp hạng các bản ghi được truy vấn. Mô hình không gian Vec-tơ được lập ra để giải quyết vấn đề khó khăn này [1].

Mô hình không gian Vec-tơ là một mô hình hiển thị thông tin dạng văn bản dưới dạng Vec-tơ. Các phần tử Vec-tơ thể hiện tầm quan trọng của một từ và sự xuất hiện của nó trong tài liệu đó [1].

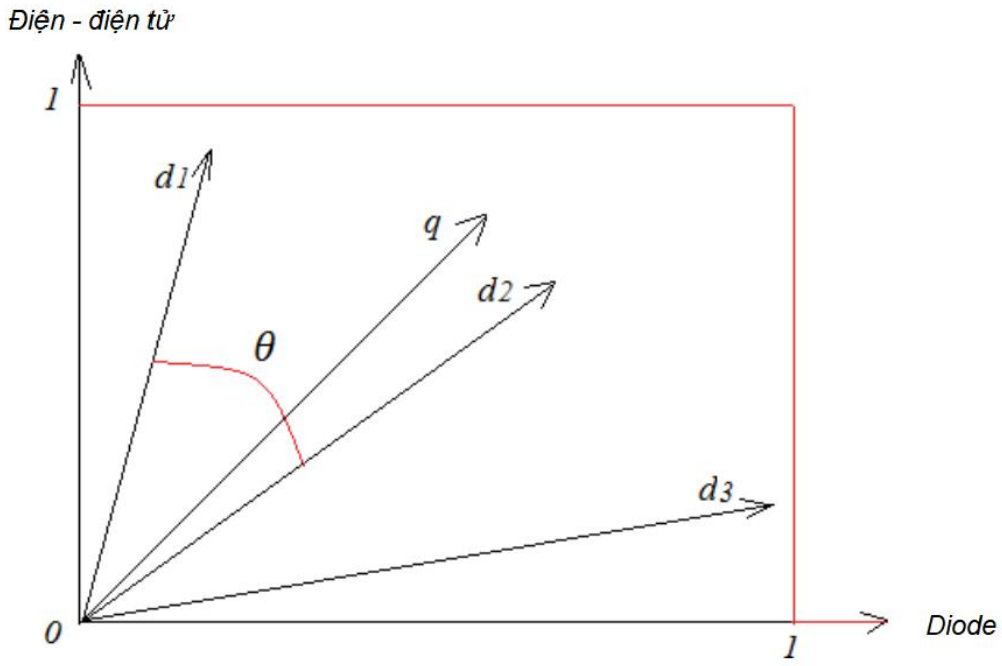
Mô hình không gian Vec-tơ hay còn gọi là **mô hình thuật ngữ Vec-tơ** (**Vec-tơ space model**) là một mô hình đại số được sử dụng để biểu diễn các tài liệu văn bản dưới dạng không gian Vec-tơ (định danh), chẳng hạn như các thuật ngữ chỉ mục. Mô hình này dùng trong hệ thống lọc thông tin (information filtering system), truy hồi thông tin, lập chỉ mục và xếp hạng mức độ phù hợp. Mô hình không gian Vec-tơ lần đầu tiên được sử dụng trong hệ thống truy hồi thông tin SMART.

Vấn đề với mô hình Boolean là không có thể tìm nạp các thành phần bị thiếu trong quá trình tính điểm với các bản ghi được truy vấn. Điều này làm

cho việc xếp hạng(ranking) trở nên khó khăn hơn, vấn đề này sẽ được giải quyết trong mô hình không gian Vec-tơ. Mô hình không gian Vec-tơ là mô hình biểu diễn thông tin văn bản dưới dạng Vec-tơ, các phần tử của Vec-tơ này thể hiện tầm quan trọng của một từ và sự xuất hiện hay không xuất hiện của nó trong một văn bản.

Ví dụ, cách biểu diễn không gian Vec-tơ với truy vấn và tài liệu

Truy vấn và tài liệu được biểu diễn bằng không gian Vec-tơ 2 chiều. Các điều khoản là *điện - điện tử* và *diode*. Có một truy vấn và ba tài liệu trong không gian Vec-tơ.



Hình 2.5: Sơ đồ ví dụ mô phỏng mô hình không gian Vec-tơ

- Tài liệu được xếp hạng cao nhất đối với các điều khoản *điện - điện tử* và *diode* sẽ là tài liệu d_2 vì góc giữa q và d_2 là nhỏ nhất. Lý do còn lại là cả *điện - điện tử* và *diode* đều nổi bật ở d_2 vì vậy nó có trọng số cao.

- Mặt khác, d_1 và d_3 cũng nhắc đến thuật ngữ nhưng trọng số không cao nên nó không là thuật ngữ quan trọng trong tài liệu.

Mô hình không gian Vec-tơ cũng biểu diễn văn bản dưới dạng các điểm trong không gian Euclide n-chiều, mỗi chiều sẽ tương ứng với một từ trong tập hợp các từ. Phần tử thứ *i*, là d_i của Vec-tơ văn bản cho biết số lần mà từ thứ *i* sẽ xuất hiện trong văn bản. Sự giống nhau của hai văn bản được định nghĩa là khoảng cách giữa các điểm, hoặc là góc giữa những Vec-tơ trong không gian với nhau [26].

Các đại lượng có trong mô hình Vec-tơ

Tần số f: là tần số xuất hiện của từ hoặc cụm từ *i* trong tài liệu *dj* được thể hiện ở công thức sau:

$$tfi,j = \frac{freqi,j}{max(freqi,j)}$$

trong đó: *freqi,j* là số lần thuật ngữ *i* xuất hiện trong tài liệu *dj*.

Tần suất nghịch đảo tài liệu: Nếu tần số xuất hiện của từ trong tài liệu cho biết mức độ phổ biến của từ đó trong hệ thống, thì việc tìm tần số nghịch đảo cho ta biết về những từ rất ít xuất hiện trong các tài liệu để có được thông tin tốt hơn. Tần suất nghịch đảo được biểu diễn:

$$idfi,j = log\frac{N}{ni}$$

Trong đó, n_i là số tài liệu chứa từ hoặc cụm *từ i, N* là tổng số tài liệu.

Mức độ tương đồng giữa 2 Vec-tơ: Để xếp hạng các tài liệu chúng ta đang có (và trả về các tài liệu có thứ hạng cao), chúng ta sẽ so sánh câu truy vấn với tập

hợp các tài liệu. Tài liệu càng gần với truy vấn thì có điểm càng cao hơn.

$$sim(d_i, q) = cos\theta$$

Trong đó θ là góc tạo bởi 2 Vec-tơ d_i, q.

Ranking: Mô hình xếp hạng tài liệu bằng cách đánh giá góc độ giữa tài liệu và câu truy vấn đó.

Ưu điểm:

- **Ranking** bằng các kết quả tương đồng.
- Các kết quả trả về trong phạm vi từ 0 đến 1 phải phù hợp với truy vấn.

Nhược điểm mô hình không gian Vec-tơ:

- Các chỉ mục độc lập với nhau nên câu truy vấn có thể làm mất ý nghĩa của câu từ.
- Mô hình không trình bày logic như thuật toán Boolean.

2.6 Đánh giá hệ thống thông qua các độ đo

Sự phù hợp, mức độ hiệu quả của bất kỳ hệ thống dữ liệu nào cũng thường được xác định thông qua các độ đo độ chính xác được mô tả sau đây.

Xét một lớp dữ liệu $c_i \in C = \{c_1, c_2, \ldots, c_m\}$ trong một bài toán phân lớp. Tập hợp các mẫu dữ liệu thuộc lớp c_i được gọi là các phần tử dương (positive). Tập hợp các mẫu dữ liệu không thuộc lớp c_i được gọi là các phần tử âm (negative). Kết quả phân lớp sau khi thực hiện phân lớp dữ liệu có thể xảy ra các trường hợp sau:

- True Positive (Trường hợp đúng dương): Phần tử dương được phân loại đúng là dương.
- False Positive (Trường hợp sai dương): Phần tử âm được phân loại sai thành dương.

- True Negative (Trường hợp đúng âm): Phần tử âm được phân loại đúng là âm.
- False Negative (Trường hợp sai âm): Phần tử dương được phân loại sai thành âm.

Ta gọi TP_i là số lượng các mẫu dữ liệu thuộc vào lớp c_i được phân loại đúng (chính xác) vào lớp c_i; gọi FP_i là số lượng các mẫu dữ liệu không thuộc lớp c_i nhưng bị phân loại sai vào lớp c_i; gọi TN_i là số lượng các mẫu dữ liệu không thuộc lớp c_i và được phân loại chính xác và gọi FN_i là số lượng các mẫu dữ liệu thuộc lớp c_i nhưng bị phân loại sai vào các lớp khác với lớp c_i.

Căn cứ vào các đại lượng trên, các khái niệm độ đo sau để đánh giá mức độ hiệu quả của hệ thống truy hồi dữ liệu:

Độ đo Precision (Mức chính xác)

Định nghĩa: Precision = TP / (TP + FP).

Ý nghĩa: Giá trị Precision càng cao thể hiện khả năng kết quả dữ liệu truy hồi được đưa ra bởi hệ thống có độ chính xác càng cao.

Độ đo Recall (Độ bao phủ, độ nhạy hoặc độ triệu hồi)

Định nghĩa: Recall = TP / (TP + FN).

Ý nghĩa: Giá trị Recall càng cao thể hiện khả năng dữ liệu đúng trong số các kết quả đưa ra của hệ thống càng cao.

Độ đo Accuracy (Độ chính xác)

Định nghĩa: Accuracy = (TP + TN) / (TP + TN + FP + FN) * 100%.

Ý nghĩa: Accuracy phản ánh độ chính xác chung của hệ thống dữ liệu.

Độ đo Specificity (Độ đặc hiệu)

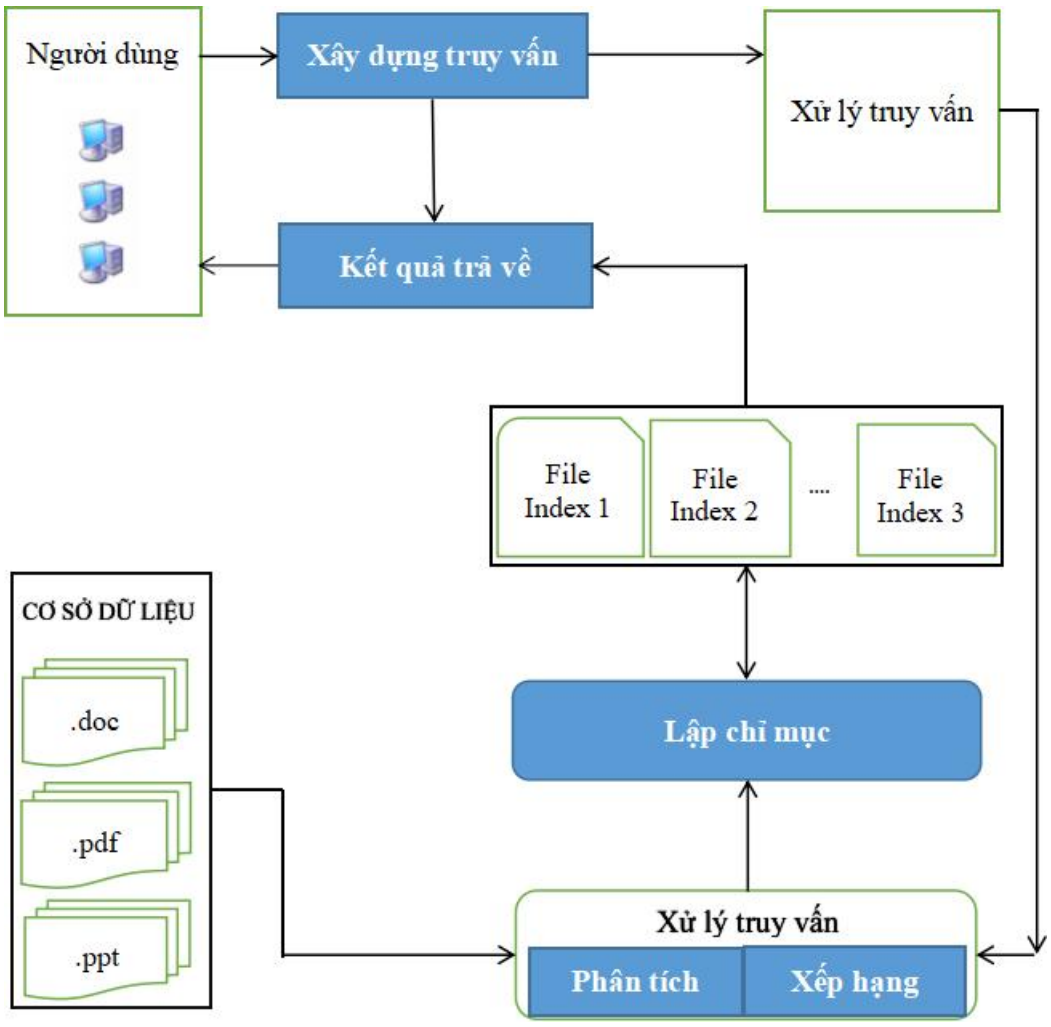
Định nghĩa: Specificity = TN/(TN+FP).

Ý nghĩa: Độ đo Specitivity đánh giá khả năng một dữ liệu là phần tử âm được hệ thống truy hồi cho ra kết quả chính xác.

Chương 3: XÂY DỰNG THỰC NGHIỆM HỆ THỐNG TRUY HỒI THÔNG TIN

Nội dung chương: Mô tả cơ chế làm việc của hệ thống truy hồi thông tin thông qua thư viện mã nguồn mở Lucene và phần chạy demo thực nghiệm hệ thống.

3.1 Mô tả hệ thống



Hình 3.1: Mô tả hệ thống truy hồi thông tin

Qua tìm hiểu các hệ thống tìm kiếm thông tin từ Google và Internet tôi nhận thấy rằng hệ thống truy hồi thông tin được mô tả qua các thành phần sau:

- Xây dựng cơ sở dữ liệu: trước tiên ta thu thập tài liệu liên quan đến chuyên ngành điện - điện tử thông qua các file văn bản .doc, .pdf, .ppt sau đó tạo thư mục chứa toàn bộ dữ liệu đó.

- Bước xử lý dữ liệu: dữ liệu sau khi được tổng hợp, ta bắt đầu phân tích, xếp hạng liên quan và đánh chỉ mục.

- Thành phần lập chỉ mục: tùy theo yêu cầu truy vấn của người dùng sẽ thực hiện đánh chỉ mục dữ liệu và đưa vào các File Index sau đó xuất kết quả trả về theo mức độ thứ hạng liên quan.

3.2 Dữ liệu

3.2.1 Loại tài liệu

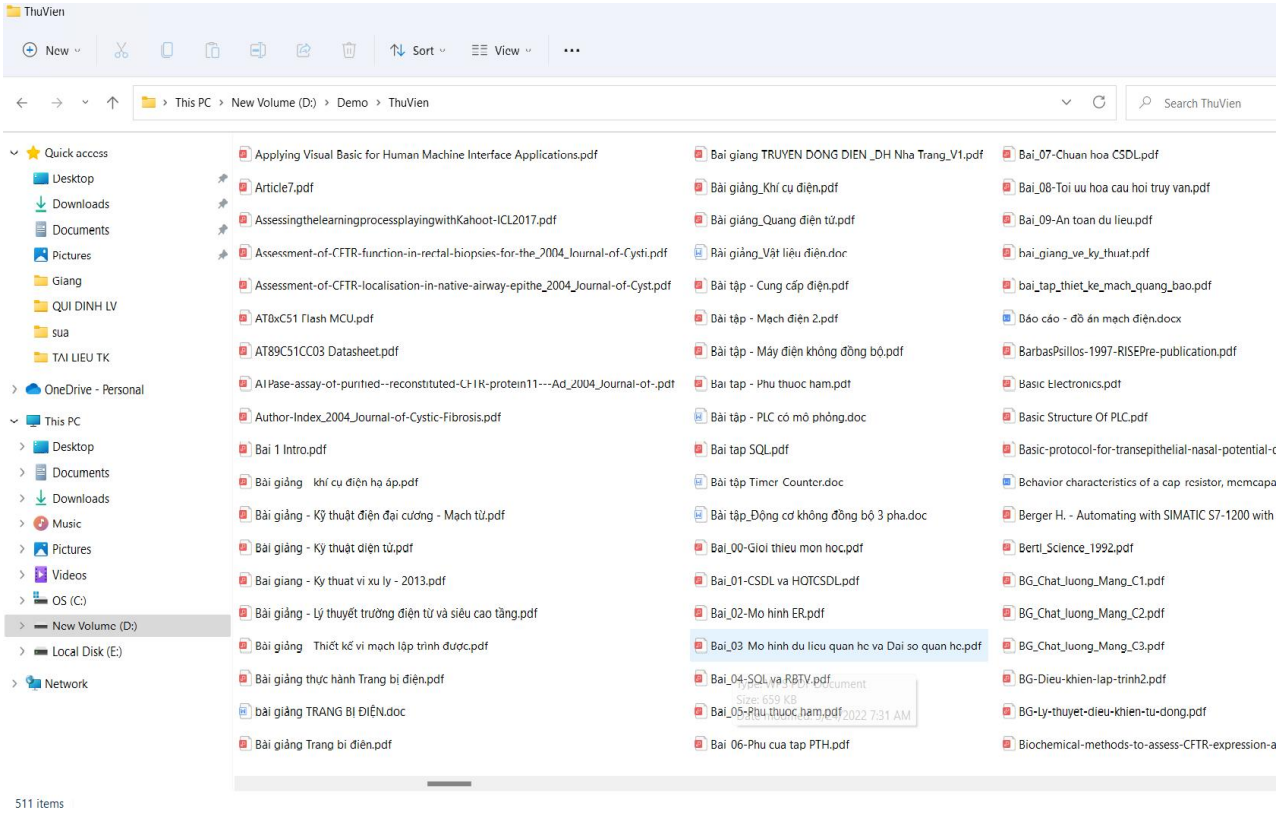
Nguồn dữ liệu được xây dựng dùng trong việc thử nghiệm là các file tài liệu chuyên ngành điện - điện tử, được thể hiện trong tập tin word, ppt, pdf.

3.2.2 Khối lượng tài liệu

Thư viện lưu trữ 511 file dữ liệu, bao gồm 220 file tài liệu tiếng việt, 291 filetiếng anh, (trong đó có 44 file tài liệu không thuộc chuyên ngành điện - điện tử mà là ngành cận liên quan như điện tử viễn thông, công nghệ thông tin).

Thư viện tài liệu được xây dựng từ thực tế giảng dạy của cá nhân và các giáo viên tại trường, còn lại là nguồn tài liệu sưu tầm từ Internet.

Tất cả các file được chứa trong thư mục ThuVien.

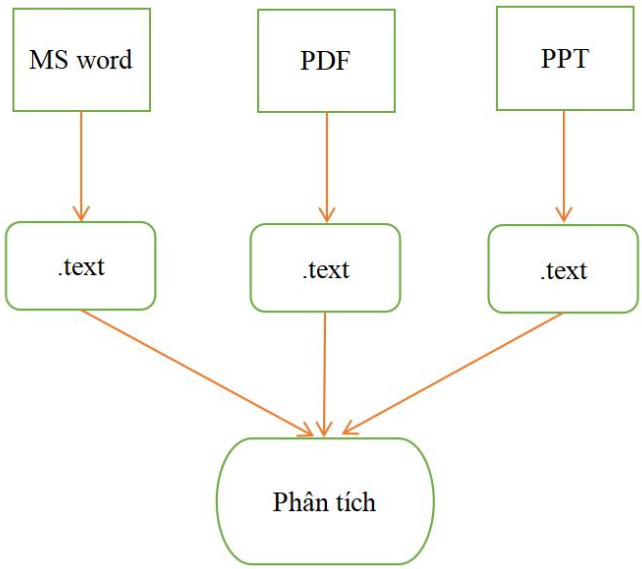


Hình 3.2: Thư viện tài liệu chuyên ngành điện - điện tử

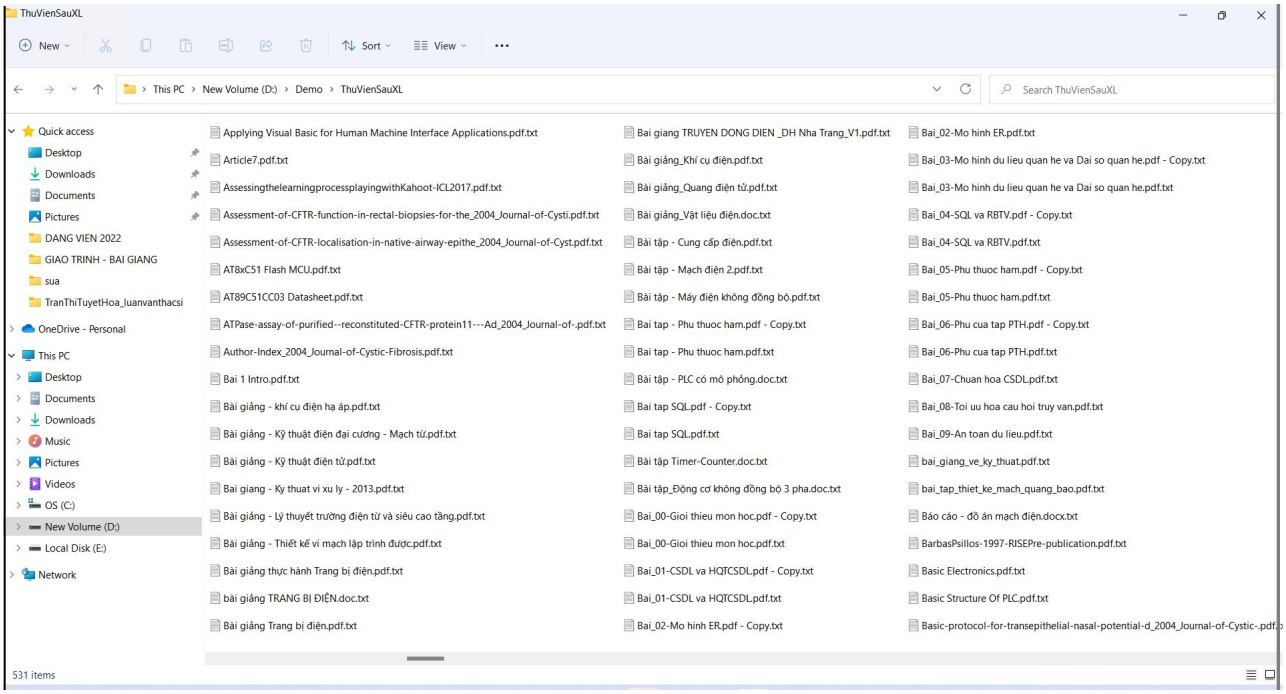
3.3 Tiền xử lý dữ liệu

- Giai đoạn xử lý dữ liệu là quá trình xử lý các dữ liệu gốc nhằm nâng cao chất lượng dữ liệu và nâng cao hiệu quả của việc tìm kiếm dữ liệu.

- Để Lucene dễ dàng phân tích và đánh chỉ mục, trước hết ta phải chuyển tài liệu về dạng văn bản thuần túy (.txt) từ những tài liệu đầu vào ở nhiều định dạng khác nhau như word, pdf, ppt…

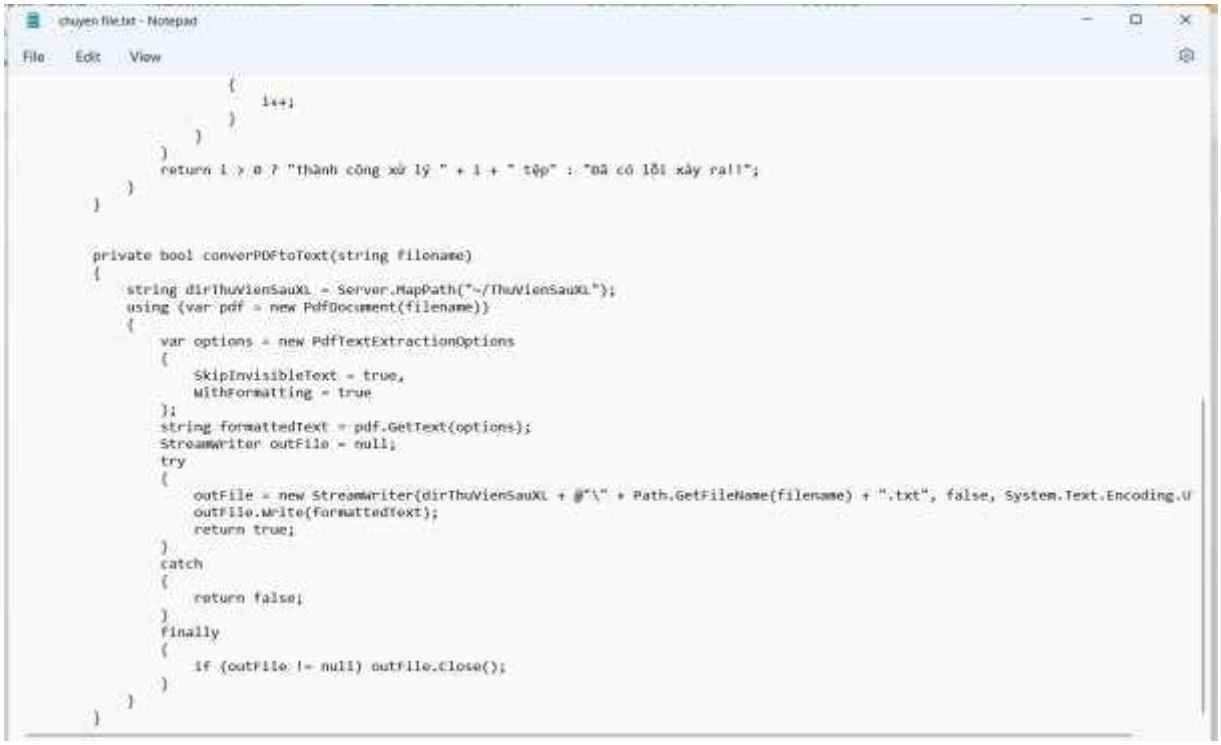


Hình 3.3: Mô hình chuyển file văn bản



Hình 3.4: Thư viện tài liệu sau khi tiền xử lý

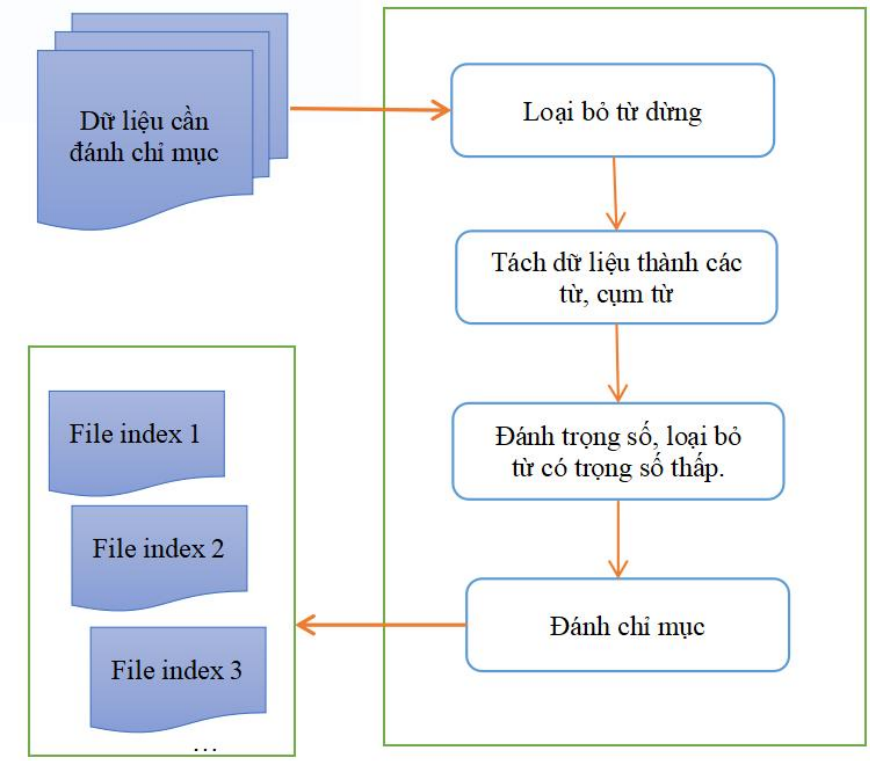
Hình 3.5: Code xử lý file sang .txt (1)



Hình 3.6: Code xử lý file sang .txt (2)

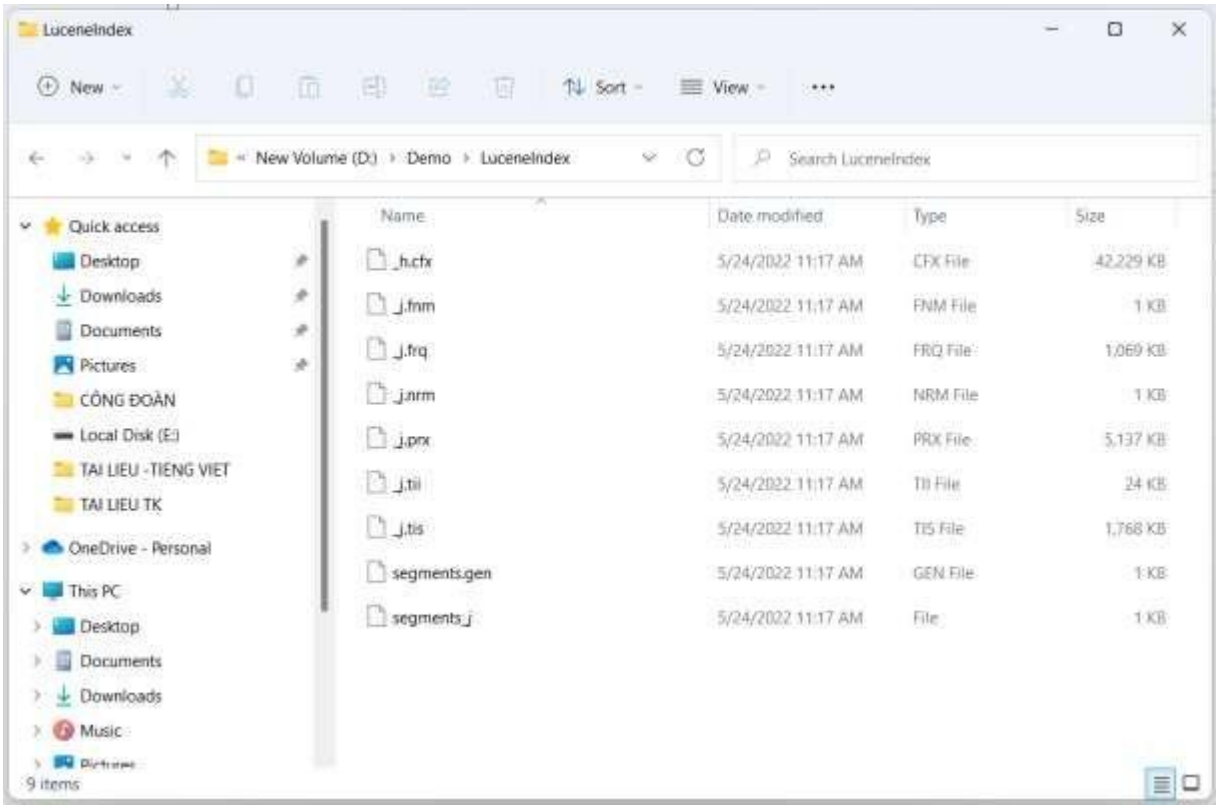
3.4 Chỉ mục Lucene

Sau khi tiền xử lý dữ liệu ta tiến hành lập chỉ mục. Để chuẩn bị cho việc lập chỉ mục, Lucene sẽ phân tích dữ liệu, phân chia dữ liệu thành các chuỗi hoặc các ký tự thông qua lựa chọn các toán tử đã thực thi hoặc loại bỏ các từ không có nghĩa, tiếp theo là đánh trọng số và loại bỏ những từ có trọng số thấp. Bên cạnh đó, thực hiện đánh chỉ mục, cập nhật file index sắp xếp theo thứ tự để dễ dàng đáp ứng nhu cầu tìm kiếm khi cần có hiệu quả cao hơn.



Hình 3.7: Quy trình lập chỉ mục Lucene

Sau khi phân tích dữ liệu, Lucene sẽ lưu dữ liệu này theo cấu trúc chỉ mục. Cấu trúc này cho phép thực hiện tìm kiếm nhanh hơn các từ khóa trong quá trình tìm kiếm.



Hình 3.8: Các tệp chỉ mục

```
Tạo Index

[HttpGet]
        public string indexLucene()
        {
            string dirIndexLucene = Server.MapPath("~/LuceneIndex");
            Lucene.Net.Store.Directory dir = Lucene.Net.Store.FSDirectory.Open(dirIndexLucene);

            //create an analyzer to process the text
            Lucene.Net.Analysis.Analyzer analyzer = new Lucene.Net.Analysis.Standard.StandardAnalyzer(Lucene.Net.Util.Version.LUCENE_30);

            //create the index writer with the directory and analyzer defined.
            Lucene.Net.Index.IndexWriter indexWriter = new Lucene.Net.Index.IndexWriter(dir, analyzer,
                      /*true to create a new index*/ true, Lucene.Net.Index.IndexWriter.MaxFieldLength.UNLIMITED);

            string dirThuVien = Server.MapPath("~/ThuVienSoURL");
            foreach (string f in Directory.EnumerateFiles(dirThuVien, "*", SearchOption.AllDirectories))
            {
                Lucene.Net.Documents.Document doc = new Lucene.Net.Documents.Document();
                string content = System.IO.File.ReadAllText(f, System.Text.Encoding.UTF8).Replace("\n", "").Replace("\r", "");
                Lucene.Net.Documents.Field fldContent =
                  new Lucene.Net.Documents.Field("content",
                  content,
                  Lucene.Net.Documents.Field.Store.YES,
                  Lucene.Net.Documents.Field.Index.ANALYZED,
                  Lucene.Net.Documents.Field.TermVector.YES);

                Lucene.Net.Documents.Field filename =
                  new Lucene.Net.Documents.Field("Filename",
                  Path.GetFileName(f),
                  Lucene.Net.Documents.Field.Store.YES,
                  Lucene.Net.Documents.Field.Index.NOT_ANALYZED);

                doc.Add(filename);
                doc.Add(fldContent);

                //write the document to the index
                indexWriter.AddDocument(doc);
            }
```

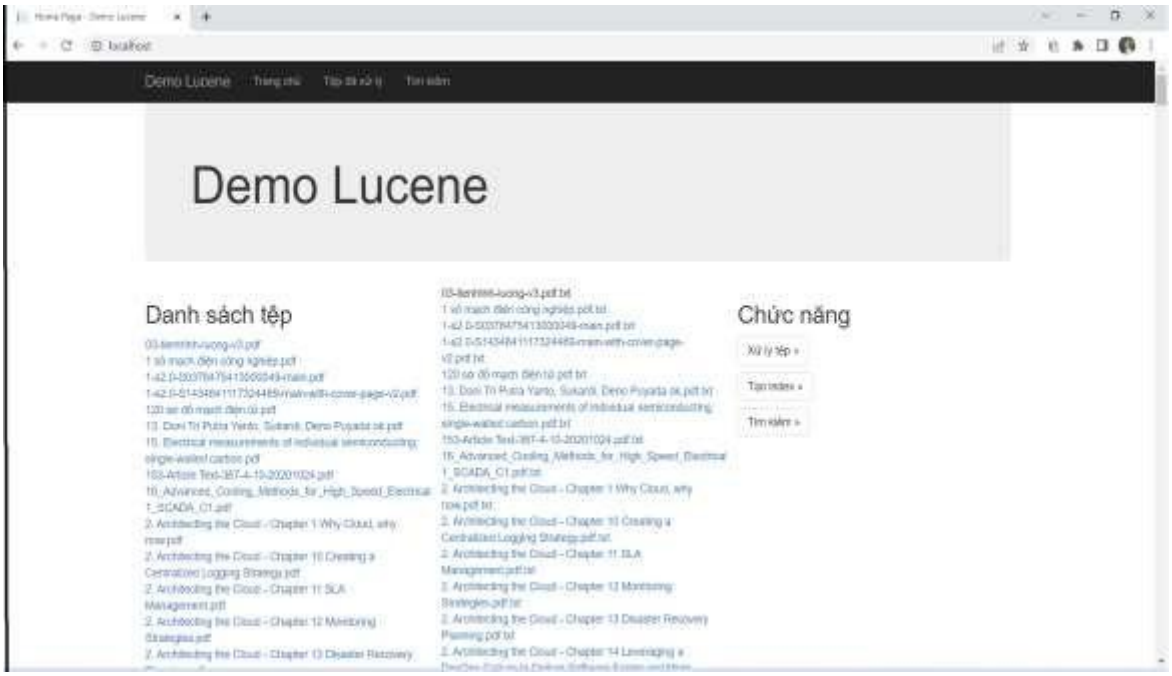
Hình 3.9: Code tạo chỉ mục

3.5. Thử nghiệm

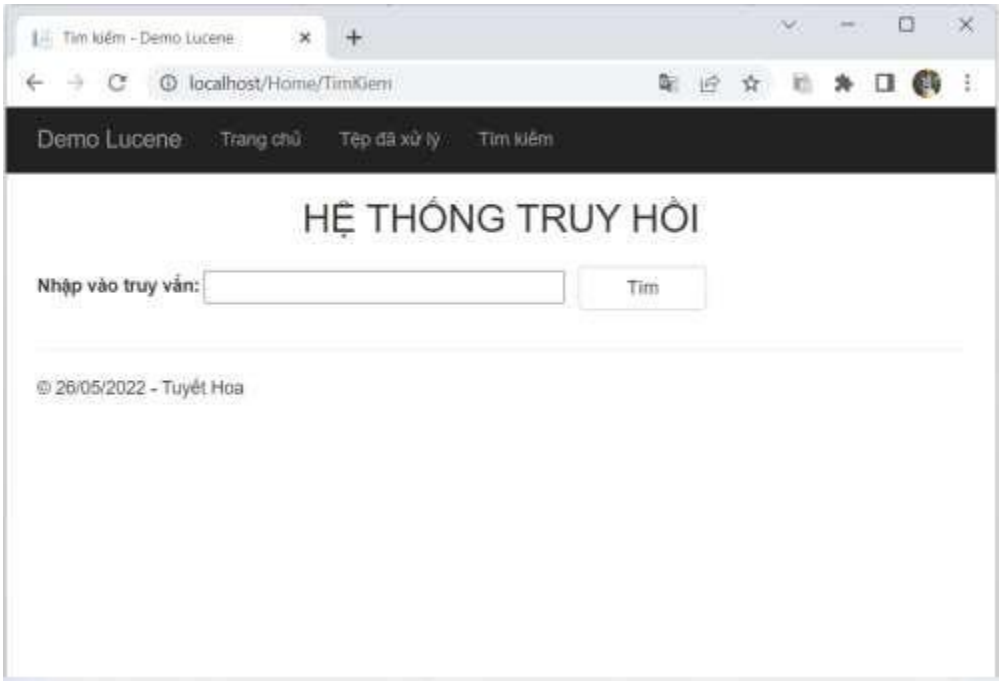
Phần trình duyệt tìm kiếm được xây dựng ở dạng giao diện Web. Cho phép người dùng nhập các từ khóa tìm kiếm theo bảng 3.1, hệ thống sẽ thực hiện tìm kiếm từ khóa trong file chỉ mục, sắp xếp kết quả và trả về danh sách các kết quả theo mức độ phù hợp giữa truy vấn và tài liệu trong cơ sở dữ liệu chỉ mục.

Bảng 3.1: Bảng từ khóa điện - điện tử sử dụng truy vấn

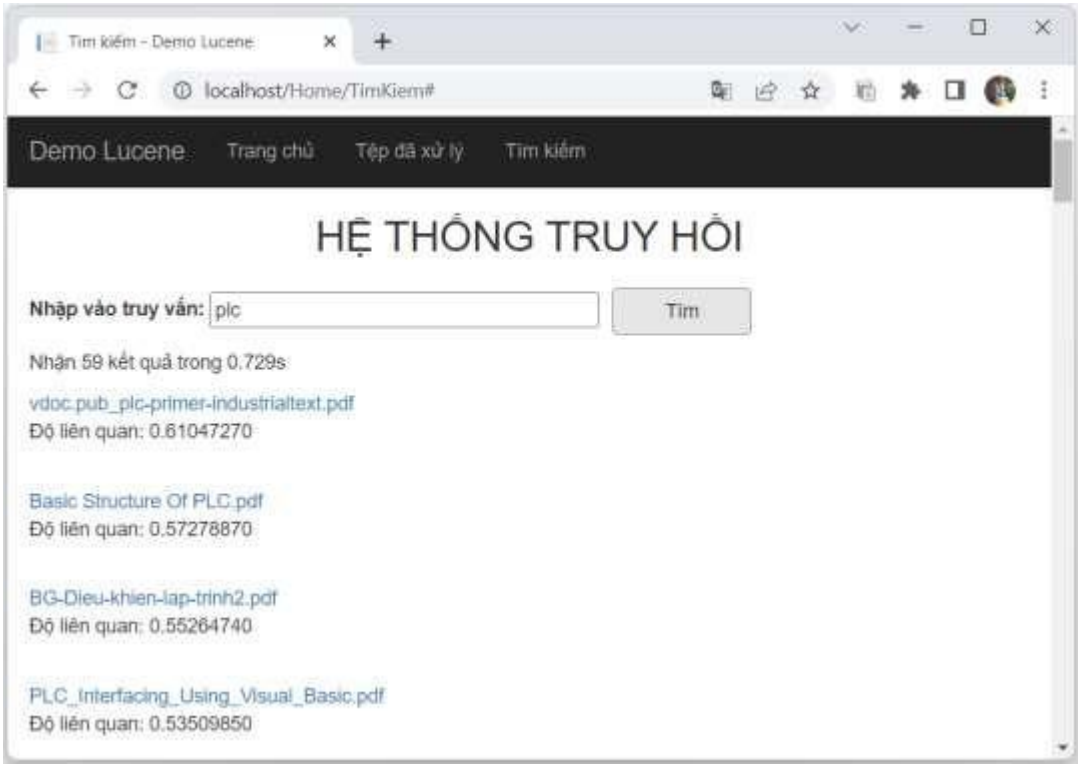
STT	Từ khóa	STT	Từ khóa
1	89C51	14	An toàn điện
2	Basic electronic	15	Biến tần
3	Circuit theory	16	Cung cấp điện
4	Diode	17	Điện tử
5	Digital electronic	18	Điện tử công suất
6	Electric machines	19	Khí cụ điện
7	Electrical measurement	20	Kỹ thuật điện
8	PLC	21	Mạch điện
9	Processor	22	Trang bị điện
10	Scada	23	Truyền động điện
11	Sensors	24	Vật liệu điện
12	Bài giảng	25	Vẽ điện
13	Giáo trình		



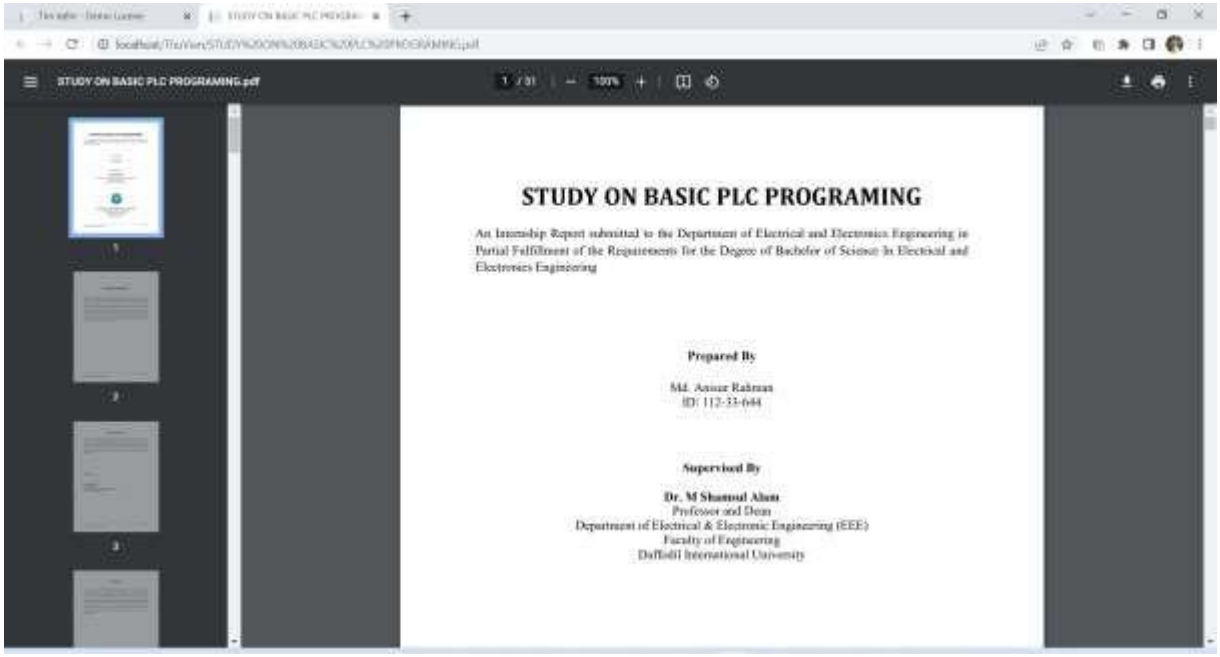
Hình 3.10: Giao diện trang chủ hệ thống tìm kiếm



Hình 3.11: Giao diện hệ thống truy hồi



Hình 3.12: Giao diện hệ thống sau khi truy hồi thông tin



Hình 3.13: Giao diện xem nội dung file tài liệu

```
[HttpPost]
        public JArray searchLucene(string content)
        {
            string dirIndexLucene = Server.MapPath("~/LuceneIndex");
            Lucene.Net.Store.Directory dir = Lucene.Net.Store.FSDirectory.Open(dirIndexLucene);

            //create an index searcher that will perform the search
            Lucene.Net.Search.IndexSearcher searcher = new Lucene.Net.Search.IndexSearcher(dir);

            JArray ar = new JArray();
            if (content.Contains(" "))
            {
                var phrase = new Lucene.Net.Search.MultiPhraseQuery();
                foreach (string s in content.Split(' ').Where(t=> t!= ""))
                {
                    phrase.Add(new Lucene.Net.Index.Term("content", s.ToLower()));
                }
                var hits = searcher.Search(phrase, 20).ScoreDocs;
                foreach (var hit in hits)
                {
                    var foundDoc = searcher.Doc(hit.Doc);
                    ar.Add(new JObject()
                    {
                        new JProperty("diem",$"{hit.Score:F8}"),
                        new JProperty("File",Path.GetFileNameWithoutExtension(foundDoc.Get("filename"))),
                    });
                }
            }
            else
            {
                Lucene.Net.Index.Term searchTerm = new Lucene.Net.Index.Term("content", content.ToLower());
                Lucene.Net.Search.Query query = new Lucene.Net.Search.TermQuery(searchTerm);
                var hits = searcher.Search(query, 20).ScoreDocs;
                foreach (var hit in hits)
                {
                    var foundDoc = searcher.Doc(hit.Doc);
                    ar.Add(new JObject()
                    {
                        new JProperty("diem",$"{hit.Score:F8}"),
                        new JProperty("File",Path.GetFileNameWithoutExtension(foundDoc.Get("filename"))),
```

Hình 3.14: Code xây dựng hệ thống tìm kiếm

3.6. Đánh giá

Trong truy hồi thông tin, độ chính xác (Precision) và độ bao phủ (Recall) được xác định theo nghĩa của một tập hợp các tài liệu được truy hồi. Ví dụ: danh sách các tài liệu trên internet có liên quan đến một chủ đề nhất định [15].

3.6.1 Độ chính xác (P)

Là tỷ lệ của các tài liệu liên quan trong tập kết quả trả về, dùng để đo lường tính chính xác của hệ thống. Nói cách khác là ước tính xem có bao nhiêu tài liệu thật sự liên quan được tìm thấy.

$$\text{Độ chính xác} = \frac{|\{\text{Tập tài liệu liên quan}\} \cap \{\text{Tập kết quả}\}|}{|\{\text{Tập kết quả trả về}\}|}$$

Ví dụ: Trong truy hồi văn bản trên một tập hợp tài liệu thì độ chính xác là số

kết quả đúng chia cho số tất cả các kết quả được trả về. Độ chính xác tính đến tất cả các tài liệu đã truy hồi tuy nhiên nó cũng được đánh giá ở một thứ hạn nhất định, chỉ xem xét các kết quả cao nhất được hệ thống trả về.

3.6.2 Độ bao phủ (R)

Là tỷ lệ của các tài liệu liên quan trong cơ sở dữ liệu tài liệu, đo lường tính toàn diện của hệ thống.

$$\text{Độ bao phủ} = \frac{|\{\text{Tập tài liệu liên quan}\} \cap \{\text{Tập kết quả}\}|}{|\{\text{Tập tài liệu liên quan}\}|}$$

Độ bao phủ còn được gọi là xác suất mà một tài liệu có liên quan được truy hồi bởi truy vấn. Khả năng trả về 100% kết quả truy vấn là rất nhỏ, do đó việc tính độ bao phủ không thì không đủ mà ta cần phải xác định thêm độ chính xác của kết quả trả về.

3.6.3 Đánh giá kết quả thực nghiệm

Để minh họa thực nghiệm sử dụng 10 câu truy vấn gồm tiếng anh và tiếng việt nhằm trải nghiệm độ tin cậy của hệ thống truy hồi.

Bảng 3.2: Thống kê độ chính xác và độ bao phủ của hệ thống (1)

STT	Truy vấn	Tài liệu tìm được	Tài liệu liên quan	P (%)	R (%)
1	Diode	47	35	41	74
2	Điện tử	64	45	57	70
3	Scada	19	15	82	79
4	PLC	59	51	62	86

5	Mạch điện	57	35	48.5	61
6	Electrical circuits	43	24	50	56
7	Vẽ điện	13	10	65	77
8	Circuit theory	27	19	60	70
9	89C51	8	5	38.8	63
10	Sensors	59	45	35.8	76
Giá trị trung bình				**54%**	**71.2%**

Trong truy hồi văn bản trên một tập hợp tài liệu thì độ chính xác là số kết quả đúng chia cho số tất cả các kết quả được trả về. Độ chính xác tính đến tất cả các tài liệu đã truy hồi tuy nhiên nó cũng được đánh giá ở một thứ hạn nhất định, vì vậy chỉ nên xem xét các kết quả cao nhất được hệ thống trả về.

Sử dụng câu truy vấn ngoài chuyên ngành điện - điện tử và bảng từ khóa tìm kiếm nội dung tài liệu liên quan trong hệ thống.

Bảng 3.3: Thống kê độ chính xác và độ bao phủ của hệ thống (2)

STT	Truy vấn	Tài liệu tìm được	Tài liệu liên quan	P (%)	R (%)
1	Pháp luật	2	0	1	0
2	Kế toán	1	0	0.8	0
3	English	8	0	10	0
4	Triết học	0	0	0	0

5	Giáo dục quốc phòng	0	0	0	0
6	Sociology	0	0	0	0
Giá trị trung bình				2%	0%

Dựa và bảng thống kê trên ta nhận thấy các truy vấn chỉ nhằm mục đích gây nhiễu hệ thống vì tỷ lệ phần trăm của độ chính xác và bao phủ là không có hoặc không đáng kể. Các tài liệu tìm được chỉ dựa vào sự trùng lặp khi so khớp với các thuật ngữ truy vấn mà không có mức độ phù hợp liên quan.

KẾT LUẬN

1. Kết quả đạt được

Bài toán giải quyết được vấn đề tìm kiếm và truy hồi thông tin mang lại hiệu quả trong việc tập trung vào dữ liệu từng lĩnh vực nhằm tránh xử lý nguồn dữ liệu lớn không liên quan.

Luận văn tiếp cận đến nghiên cứu các vấn đề về truy hồi thông tin, các đánh giávề hệ truy hồi thông tin giúp xác định khả năng tự tìm kiếm của truy hồi thông tin; nghiên cứu và các kỹ thuật lập chỉ mục mô hình truy hồi thông tin; và phân loại văn bản dựa vào kỹ thuật máy học (Machine learning techniques). Từ đó thực hiện bài toán "Xây dựng hệ thống truy hồi học liệu cho sinh viên ngành điện – điện tử".

Luận văn cũng tìm hiểu một cách hệ thống các tính năng và hoạt động của mã nguồn mở Lucene như: Lucene cung cấp khả năng phân tích dữ liệu, tiền xử lý, tạo chỉ mục cho các tài liệu để xây dựng nên hệ thống chỉ mục, cung cấp khả năng tiếp nhận cáccâu truy vấn của người dùng, thực hiện tìm kiếm dựa trên hệ thống chỉ mục đã có và truy hồi kết quả tìm kiếm.

2. Hạn chế

Bên cạnh những kết quả đạt được thì luận văn cũng có những mặc hạn chế như sau:

- Phần trình bày các nội dung của luận văn tương đối hạn chế dẫn đến tính thuyết phục của bài toán chưa cao.

- Khả năng áp dụng phương pháp IF và IDF để đánh trọng số và xếp hạng liên quan của tài liệu với truy vấn chưa phân tích hết mức độ liên quan của từng thuật ngữ trong tài liệu mà chỉ dựa trên số lần của từ xuất hiện trên văn bản, dẫn đến kết quả độ chính xác và độ bao phủ chưa cao.

3. Hướng phát triển

- Tìm hiểu các cơ sở lý thuyết liên quan và các kỹ thuật học máy ứng dụng trong giải pháp giải quyết bài toán mang tính thuyết phục cao.

- Áp dụng kết hợp kỹ thuật phân tích ngữ nghĩa tiềm ẩn (LSA) trong tự nhiên và lập chỉ mục ngữ nghĩa tiềm ẩn (LSI) với phương pháp TF và IDF trong việc đánh trọng số và lập chỉ mục để mang lại kết quả tìm kiếm truy hồi dữ liệu có độ chính xác hơn.

Thông qua cơ sở lý thuyết và bài toán thực nghiệm, tôi sẽ đề xuất áp dụng đề tài vào thực tế tại trường trung cấp kinh tế kỹ thuật Tây Ninh nơi tôi đang công tác, và có thể phát triển, thay đổi hệ thống để đưa ra khả năng tìm kiếm thông tin tốt nhất. Khi đó đề tài không những chỉ áp dụng cho sinh viên ngành điện – điện tử mà có thể áp dụng cho tất cả các ngành nghề đào tạo tại trường hay ở những trường học khác nhằm đáp ứng nhu cầu học tập của sinh viên học sinh.

DANH MỤC CÁC TÀI LIỆU THAM KHẢO

[1] Arman Rasool Faridi, "Trends and issues in Modern Information Retrieval", Department of Computer Science, Aligarh Muslim University, Aligarh, arman.faridi@gmail.com Aasim Zafar, Department of Computer Science, Aligarh Muslim University, Aligarh, aasimzafar@gmail.com

[2] Bhaskar Mitra, Microsoft, "An Introduction to Neural Information Retrieval", University College London, Montreal, Canada bmitra@microsoft.com and Nick Craswell Microsoft Bellevue, USA nickcr@microsoft.com Suggested Citation: Bhaskar Mitra and Nick Craswell (2018), Vol. xx, No. xx, pp 1–18. DOI: 10.1561/XXXXXXXXX.

[3] Christopher D. Manning Prabhakar Raghavan Hinrich Schütze, "An Introduction to Information Retrieval", Cambridge University Press Cambridge, England.

[4] Gabriella Pasi**,** "Intelligent Information Retrieval: some research trends", Istituto per le Tecnologie della Costruzione Sezione Tecnologie Informatiche Multimediali Consiglio Nazionale delle Ricerche via Ampère, 56, 20131 – Milano e-mail: gabriella.pasi@itim.mi.cnr.it.

[5] Osman Ali Sadek Ibrahim, "Evolutionary Algorithms and Machine Learning Techniques for Information Retrieval", ASAP Research Group School of Computer Science The University of Nottingham United Kingdom September, 2017.

[6] G. Desjardins and R. Godin, "Performance of Information Retrieval Models Using Term Co-occurrences", *Department of computer science* & R. Proulx, *Department of psychology* University of Quebec in Montreal, Canada.

[7] https://www.https://www.researchgate.net/publication/303806260_Machine_Learning_Algorithms_and_Applications, truy cập ngày 10/8/2021

[8] http://www.dcs.gla.ac.uk/Keith/Chapter.1/Ch.1.html, truy cập ngày 10/8/2021

[9] https://en.wikipedia.org/wiki/Information_retrieval, truy cập ngày 15/8/2021

[10] https://www.tutorialspoint.com/natural_language_processing/natural_language_processing_information_retrieval.htm, truy cập ngày 12/10/2021

[11] https://kipalog.kaopiz.com/posts/Lucene, truy cập ngày 12/03/2022

[12] https://lucene.apache.org, truy cập ngày 12/03/2022

[13] https://www.tutorialspoint.com/lucene/lucene_indexing_process.htm, truy cập ngày12/03/2022

[14] https://github.com/isoboroff/trec-demo, truy cập ngày 12/03/2022

[15] https://www.kaggle.com/datasets/atamazian/sklearndeltatfidf, truy cập ngày 12/10/2021

[16] https://en.wikipedia.org/wiki/Precision_and_recall, truy cập ngày 30/04/2022

[17] https://scikitlearn.org/stable/auto_examples/model_selection/plot_precision_recall.html, truy cập ngày 30/04/2022

[18] https://blog.duyet.net/2019/08/ir-evaluation.html, truy cập ngày 05/03/2022

[19] https://helpex.vn/article/tim-kiem-va-lap-chi-muc-voi-apache-lucene/, truy cập ngày 05/03/2022

[20] https://viblo.asia/p/phan-loai-van-ban-tu-dong-bang-machine-learning/, truy cập ngày 20/03/ 2022 [21]

[21] https://vi.wikepedia.org/wiki/Mô-hình-không-gian-vectơ/, truy cập ngày 25/03/2022

[22] https://nguyenvanhieu.vn/phan-loai-van-ban-tieng-viet/, truy cập ngày

20/03/2022

[23] http://trituevietvn.com/chi-tiet/-Phan-mem-quan-ly-ho-so-tim-kiem-theo-noi-dung-dung-Lucene-18,truy cập ngày 25/03/2022

[24] https://www.researchgate.net/publication/235907860_Phat_Trien_he_truy_hoi_thong_tin_tieng_Viet_dua_tren_ma_nguon_mo_Vietnamese_language_information_retrie val_using_open_source/, truy cập ngày 27/03/2022

[25] https://tailieu.vn/doc/tom-tat-luan-van-thac-si-nganh-cong-nghe-thong-tin-nghien-cuu-cong-nghe-tim-kiem-ma-nguon-mo-luce-2075493.html, truy cập ngày 28/03/2022

[26] https://123docz.net//document/2399619-ung-dung-giai-thuat-di-truyen-vao-phan- loai-tai-lieu-dang-van-ban.htm, truy cập ngày 28/03/2022

BẢN CAM ĐOAN

Tôi cam đoan đã thực hiện việc kiểm tra mức độ tương đồng nội dung luận văn/luận án qua phần mềm Kiểm tra tài liệu một cách trung thực và đạt kết quả mức độ tương đồng **17%** toàn bộ nội dung luận văn/luận án. Bản luận văn/luận án kiểm tra qua phần mềm là bản cứng luận văn đã nộp để bảo vệ trước hội đồng. Nếu sai tôi xin chịu các hình thức kỷ luật theo quy định hiện hành của Học viện.

TP.Hồ Chí Minh, ngày 15 tháng 07 năm 2022

Học viên cao học

Trần Thị Tuyết Hoa

Kết quả kiểm tra trùng lặp #110855 KiemTraTaiLieu

KiểmTraTàiLiệu

BÁO CÁO KIỂM TRA TRÙNG LẶP

Thông tin tài liệu

Tên tài liệu:	TranThiTuyetHoa_luanvanthacsi
Tác giả:	Trần Thị Tuyết Hoa
Điểm trùng lặp:	17
Thời gian tải lên:	08:49 15/07/2022
Thời gian sinh báo cáo:	08:52 15/07/2022
Các trang kiểm tra:	Trang 5-67

Kết quả kiểm tra trùng lặp

17% Có 17% nội dung trùng lặp

83% Có 83% nội dung không trùng lặp

0% Có 0% nội dung người dùng loại trừ

0% Có 0% nội dung hệ thống bỏ qua

Nguồn trùng lặp tiêu biểu

123docz.net tailieu.vn vi.wikipedia.org

Báo cáo tạo lúc 08:52 15/07/2022 tại https://app.kiemtratailieu.vn Trang 1 / 23

Học viên **Người hướng dẫn khoa học**

Trần Thị Tuyết Hoa **TS. Tân Hạnh**