XÂY DỰNG KHO DỮ LIỆU PHỤC VỤ HỆ THỐNG PHÁT HIỆN SAO CHÉP

BUILDING A DATA WAREHOUSE FOR DUPLICATE DETECTION SYSTEM

Châu Thùy Dương[1], Võ Trung Hùng[2]*, Hồ Phan Hiếu[2]**

[1]Trường Đại học Quảng Nam; chauthuyduong.qn@gmail.com
*[2]Đại học Đà Nẵng; *vthung@dut.udn.vn, **hophanhieu@gmail.com*

Tóm tắt - Trong bài báo này, chúng tôi trình bày kết quả xây dựng kho dữ liệu sẽ được sử dụng trong hệ thống phát hiện sao chép từ các nguồn tài liệu của Đại học Đà Nẵng. Kho dữ liệu này bao gồm các tài liệu gốc, cơ sở dữ liệu thông tin chung về tài liệu và dữ liệu đã được trích xuất từ các tài liệu lưu trữ dưới định dạng XML. Chúng tôi đã đề xuất cấu trúc lưu trữ và các chương trình tương ứng để dễ dàng lưu trữ, cập nhật kho dữ liệu và xử lý các dữ liệu trên kho. Chúng tôi đã tiến hành thử nghiệm và lưu trữ trên kho này với hơn 100 tài liệu mỗi loại cho báo cáo tốt nghiệp của sinh viên ngành công nghệ thông tin, luận văn tốt nghiệp cao học và báo cáo tổng kết đề tài nghiên cứu. Kết quả nghiên cứu này là tiền đề để xây dựng một hệ thống phát hiện tự động việc sao chép trái phép trên các tài liệu khoa học, góp phần hạn chế nạn "đạo văn" đang diễn ra phổ biến hiện nay.

Từ khóa - phát hiện sao chép; kho dữ liệu; đạo văn; chuyển định dạng; học liệu.

Abstract - This paper presents the results of building a data bank to be used in the duplication detection system from learning resources of the University of Danang. This data bank includes original documents, abstract information database about the original documents and the data which has been extracted from the documents to store in XML format. We have proposed storage structure and the corresponding programs to easily store, update and manage data in data bank. We have experimented and stored in this data bank over 300 documents such as course papers by IT students, master theses and reports of research projects. Results of this study imply a prerequisite for building an automated system to detect the duplication in the scientific documents, contributing to controlling "plagiarism".

Key words - duplication detection; data bank; plagiarism; format converting; learning resource.

1. Giới thiệu

Ứng dụng công nghệ thông tin trong dạy và học đang là một xu hướng tất yếu và mang lại hiệu quả cao. Một trong những ứng dụng quan trọng nhất là trao đổi thông tin, tài liệu thông qua môi trường Internet. Hiện tại, tài liệu trên Internet đang dần trở thành là nguồn tham khảo chính và không thể thiếu đối với người dạy và người học.

Tuy nhiên, cùng với sự phổ biến của việc tham khảo tài liệu trên mạng Internet, tình trạng "đạo văn" cũng đang có xu hướng gia tăng và đã đến mức báo động trong những năm gần đây. Tình trạng sinh viên trong các trường đại học sao chép các đồ án, luận văn tốt nghiệp trên mạng Internet của những khóa trước ngày càng nhiều, dần trở nên phổ biến và trở thành một vấn nạn làm suy giảm chất lượng đào tạo.

Làm thế nào để hạn chế tình trạng "đạo văn" đang là một câu hỏi lớn đặt ra cho toàn xã hội.

Một trong những giải pháp để hạn chế tình trạng trên là xây dựng các phần mềm nhằm phát hiện và chỉ ra những nội dung nào trên một tài liệu là được sao chép lại từ những tài liệu đã có trước đó và mức độ sao chép như thế nào. Việc phát hiện này vừa có tác dụng giúp cho chính các tác giả kiểm tra và điều chỉnh văn bản của mình, vừa có tác dụng giúp giáo viên, các nhà quản lý phát hiện sự gian lận trên văn bản cần kiểm tra.

Để xây dựng được một phần mềm như vậy, trước hết cần phải có một kho dữ liệu lưu trữ các tài liệu đã có trước đó và tiếp đến cần phải có các phương pháp, giải thuật để phát hiện và đánh giá các nội dung sao chép từ các tài liệu đã lưu trữ trên kho . Kho dữ liệu càng nhiều thì càng có lợi cho việc phát hiện sao phép vì độ bao phủ của nó càng lớn.

Trong bài báo này, chúng tôi trình bày lại kết quả xây dựng kho dữ liệu sẽ được sử dụng trong hệ thống phát hiện sao chép từ các nguồn tài liệu của Đại học Đà Nẵng. Kho dữ liệu này bao gồm các tài liệu gốc (luận án tiến sĩ, luận văn tốt nghiệp thạc sĩ, báo cáo đồ án/luận văn tốt nghiệp đại học, các báo cáo tổng kết đề tài nghiên cứu khoa học và các tài liệu khác), cơ sở dữ liệu thông tin chung về tài liệu và dữ liệu đã được trích xuất từ các tài liệu lưu trữ dưới định dạng XML. Chúng tôi đề xuất một cấu trúc lưu trữ và các chương trình tương ứng để dễ dàng lưu trữ, cập nhật kho dữ liệu và xử lý các dữ liệu trên kho. Chúng tôi đã tiến hành thử nghiệm và lưu trữ trên kho này với hơn 100 tài liệu mỗi loại cho báo cáo tốt nghiệp của sinh viên ngành công nghệ thông tin, luận văn tốt nghiệp cao học và báo cáo tổng kết đề tài nghiên cứu.

2. Hệ thống phát hiện sao chép

Cho một văn bản `D` gọi là văn bản kiểm tra và `M` là tập hợp văn bản nguồn đã được đăng ký trước, bài toán đặt ra là xác định độ tương tự của văn bản `D` với từng văn bản `m` trong `M`. Nếu độ tương tự của `D` với các văn bản trong `M` vượt quá một ngưỡng nào đó thì `D` được coi là sao chép từ các văn bản `M`.

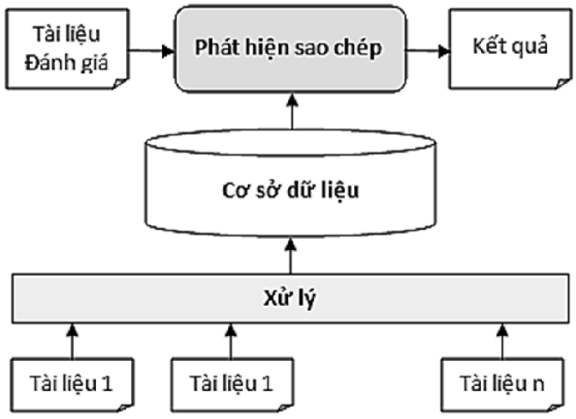
Hệ thống phát hiện sao chép là hệ thống xác định vị trí trùng lặp và đo độ tương tự giữa các tài liệu. Việc đo độ tương tự giữa hai tài liệu thường dựa trên việc đo độ tương tự giữa thành phần đơn vị trong văn bản kiểm tra với thành phần đơn vị trong văn bản nguồn. Việc phân biệt các phương pháp phát hiện sao chép dựa trên phương pháp đó xác định các thành phần hay đơn vị khác nhau giữa các văn bản như thế nào (các thành phần đơn vị này có thể là từ, câu, đoạn hoặc toàn bộ văn bản).

Mô hình tổng quát của một hệ thống phát hiện sao chép, Hình 1.

Để phát hiện việc sao chép (nếu có) trên tài liệu đánh giá từ các tài liệu đã có, người ta thường dùng một số phương pháp như sau:

Cops: được phát triển vào năm 1995 bởi Brin, Davis và

Garcia Molina [1]. Cops thực hiện so sánh giữa văn bản đánh giá và tập các văn bản huấn luyện theo đơn vị câu. Các câu được so sánh với nhau dựa theo giá trị băm của chúng. Nếu số câu giống nhau giữa văn bản kiểm tra và các văn bản trong tập huấn luyện vượt ngưỡng cho trước thì kết luận có sao chép, ngược lại là không sao chép. Cops có ba nhược điểm: thứ nhất là sự va chạm trong phương thức băm, nhiều câu khác nhau có cùng giá trị băm. Thứ hai, Cops cho kết quả rất tốt với những câu hoàn toàn giống nhau nhưng nó không thể phát hiện khi chúng giống nhau một phần. Thứ ba, Cops sử dụng đơn vị câu để phát hiện sao chép nên phụ thuộc nhiều vào việc tách câu.



Hình 1. *Mô hình tổng quát hệ thống phát hiện sao chép*

Scam: được phát triển vào năm 1996 bởi Shivakumar nhằm cải thiện Cops [2]. Scam dựa trên kỹ thuật tìm kiếm và thu hồi thông tin kết hợp với mô hình không gian véc-tơ để giải quyết việc va chạm giá trị băm. Scam phát hiện sao chép dựa trên đơn vị từ. Mỗi tài liệu được coi như là một véc-tơ từ vựng trong toàn bộ tập hợp đang xét, giá trị mỗi phần tử trong véc-tơ là tỉ lệ xuất hiện của từ vựng trong văn bản đó. Scam có thể phát hiện việc trùng lặp một phần nhưng nó có thể cho những kết quả chứa những khẳng định giả khi so sánh các tài liệu chỉ dựa trên từ vựng. Scam không thể cung cấp thông tin vị trí trùng lặp giữa những tài liệu. Một điểm yếu khác của Scam là độ tương tự không được định nghĩa rõ ràng để có thể chọn một ngưỡng cho nhiều loại tài liệu.

Koala: dựa trên việc lựa chọn tập hợp các đơn vị của các ký tự và tính toán độ tương tự dựa trên giá trị băm của các đơn vị này. Mức độ giống nhau giữa hai tài liệu được đo bằng cách đếm số lượng các đơn vị chung trong các tài liệu. Khó khăn của kỹ thuật này là độ chính xác phụ thuộc rất lớn vào việc lựa chọn các đơn vị trong tài liệu [3] [4].

Check: sử dụng đoạn làm đơn vị so sánh. Trích xuất thông tin có cấu trúc và từ khóa từ các tài liệu, sử dụng chúng để kiểm tra sự chồng chéo lẫn nhau. Check chỉ giới hạn cho tài liệu có cấu trúc [5].

Hầu hết những phương pháp trên sử dụng mô hình không gian véc-tơ hoặc hàm `cosin` để tính độ tương tự. Tuy nhiên nó chỉ giới hạn trong việc tính toán mức độ sao chép. Một số nhà nghiên cứu sau này đã đưa ra phương pháp đo độ tương đồng trực quan theo hàm `Sim()` như sau:

Theo [6], đưa vào hai tài liệu d_1 và d_2, cho $|d_1 \cap d_2|$ là số câu chung của hai tài liệu d_1 và d_2. Cho $|d_1|$ là số lượng câu trong tài liệu d_1, $|d_2|$ là số lượng câu trong tài liệu d_2 thì $sim(d_1 \cap d_2)$ là độ tương tự giữa tài liệu d_1 và d_2:

$$sim(d_1 \cap d_2) = \left(\frac{|d_1 \cap d_2|}{|d_1|}, \frac{|d_1 \cap d_2|}{|d_2|} \right)$$

Số hạng đầu tiên tính số lượng câu chung của d_1 và d_2 chia cho số lượng câu của d_1, giá trị này đại diện cho một phần d_1 chứa trong d_2. Số hạng thứ hai tính số lượng câu chung của d_1 và d_2 chia cho số lượng câu d_2, nó đại diện cho phần d_2 chứa trong d_1.

Ví dụ: Xét hai tài liệu d_1 và d_2, với d_1 chứa 120 câu, d_2 chứa 160 câu, d_1 và d_2 có 80 câu chung. Sau khi so sánh, $sim(d_1 \cap d_2)$ trả về cặp giá trị (0,667; 0,500) là mức độ giống nhau giữa d_1 và d_2. Nó đại diện cho mức độ tương đồng, hai phần ba số câu trong d_1 được tìm thấy trong d_2 và một nửa số câu d_2 được tìm thấy trong d_1.

$sim(d_1 \cap d_2)$ về bản chất đo lường mối quan hệ con giữa d_1 và d_2. Vì vậy mỗi số hạng được đặt giữa 0 và 1. Trường hợp đặc biệt, nếu d_1 và d_2 đồng nhất thì giá trị $sim(d_1 \cap d_2)$ là (1,00;1,00). Nếu d_1 và d_2 không có câu chung thì $sim(d_1 \cap d_2)$ có giá trị là (0;0).

Sự tương ứng này đến mối quan hệ tập con giải thích tại sao độ đo tương đồng được trả về một cặp thay cho một số đơn lẻ. Trường hợp tổng quát, cho A và B là hai bộ không rỗng, A có quan hệ chứa trong B ($A \subset B$). Mặc dù A là tập con khác rỗng của B, nhưng không có lý do để khẳng định B cũng được chứa trong A và kích thước của B liên quan đến A không được xác định. Vì vậy cần hai số hạng để giữ thông tin về quan hệ của A với B và quan hệ của B với A. Giảm số hạng trong cặp có thứ tự hoặc kết hợp số hạng vào trong một số duy nhất sẽ dẫn đến làm mất thông tin [7].

3. Phân tích, thiết kế kho dữ liệu

3.1. Khảo sát dữ liệu

Để thiết kế kho dữ liệu phục vụ hệ thống phát hiện sao chép từ các nguồn tài liệu của Đại học Đà Nẵng (ĐHĐN), chúng tôi đã tiến hành khảo sát một số tài liệu sau:

1. Báo cáo đồ án, luận văn tốt nghiệp của sinh viên

Đây là loại báo cáo có số lượng lớn (mỗi năm ĐHĐN có khoảng 10.000 báo cáo tốt nghiệp) và rất khó kiểm soát việc các báo cáo đó có chứa các nội dung sao chép không hợp lệ từ các nguồn khác hay không.

Về mặt nội dung, mỗi báo cáo có chứa các nội dung chính gồm: thông tin trên trang bìa và bìa phụ; lời cảm ơn; lời cam đoan; nhận xét của giáo viên hướng dẫn; nhận xét của hội đồng bảo vệ; mục lục; danh mục từ viết tắt (nếu có); danh mục hình vẽ; danh mục bảng; mở đầu; các chương 1, 2, 3, …; kết luận và hướng phát triển; phụ lục (nếu có), tài liệu tham khảo và tóm tắt luận văn.

2. Báo cáo luận văn tốt nghiệp thạc sĩ

Đây là loại báo cáo có số lượng khá lớn (mỗi năm có khoảng 2.000 báo cáo tốt nghiệp) và về mặt bố cục thì tương tự với báo cáo tốt nghiệp của sinh viên. Một điểm đáng lưu ý với loại báo cáo này khả năng sao chép từ các nguồn tài liệu khác ngoài ĐHĐN. Vì vậy, khi xây dựng kho dữ liệu phải chú ý thu thập cả các báo cáo tốt nghiệp của học viên ngoài ĐHĐN.

3. Báo cáo tổng kết các đề tài khoa học

Mỗi năm, ĐHĐN có khoảng 200 báo cáo tổng kết các đề tài nghiên cứu khoa học. Về bố cục cũng gần giống như 2 loại tài liệu trên nhưng có thêm các thông tin về đơn vị chủ quản, đơn vị thực hiện, loại đề tài,…

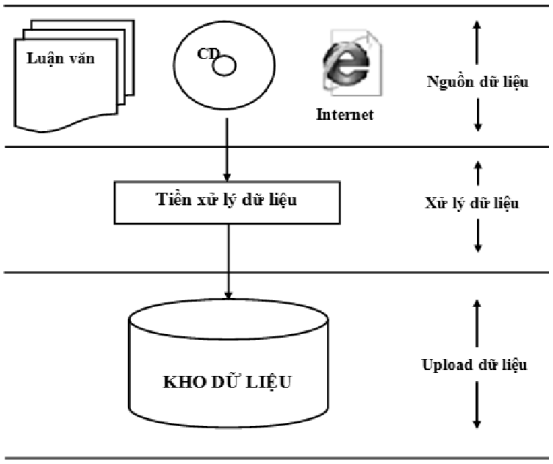
4. Bài báo từ Tạp chí Khoa học và Công nghệ

Mỗi năm, ĐHĐN xuất bản 12 số với khoảng 15-20 ấn bản (có số in nhiều ấn bản) và số lượng các bài báo khoảng 300-500 bài. Về thông tin thì mỗi bài báo thường có nhiều tác giả và nội dung mỗi bài chỉ 6-10 trang.

Trong kho dữ liệu này, ở bước đầu chúng tôi tập trung lưu trữ và xử lý các loại tài liệu như mô tả ở trên. Trong giai đoạn kế tiếp, sẽ mở rộng lưu trữ và xử lý các tài liệu từ các đơn vị khác và các tài liệu trên mạng Internet.

3.2. Qui trình

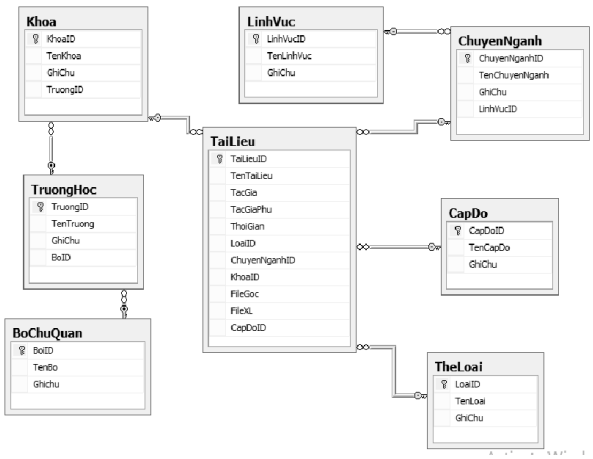
Trên cơ sở khảo sát, chúng tôi đề xuất một qui trình để xây dựng kho dữ liệu như sau:



Hình 2. *Qui trình xây dựng kho dữ liệu*

3.3. Thiết kế hệ cơ sở dữ liệu

Trên cơ sở khảo sát dữ liệu và chọn lọc các thông tin cần thiết phải lưu trữ, chúng tôi đề xuất lưu trữ dữ liệu theo mô hình dữ liệu quan hệ như sau:



Hình 3. *Mô hình cơ sở dữ liệu quan hệ*

Trong đó, các bảng dữ liệu gồm:

- **`Bochuquan`**: lưu trữ thông tin của Bộ quản lý.
- **`Truonghoc`**: lưu trữ thông tin của trường thuộc một bộ chủ quản.
- **`Khoa`**: lưu trữ thông tin của khoa thuộc một trường nào đó.
- **`Linhvuc`**: lưu trữ thông tin về lĩnh vực đào tạo hoặc nghiên cứu khoa học.
- **`Chuyennganh`**: lưu trữ thông tin về chuyên ngành thuộc lĩnh vực của tài liệu, ví dụ như khoa học máy tính, kế toán, xây dựng, …
- **`Theloai`**: lưu trữ thông tin loại tài liệu, ví dụ tài liệu thuộc thể loại luận văn thạc sĩ, đồ án tốt nghệp, …
- **`Capdo`**: lưu trữ thông tin về cấp độ quan trọng của loại tài liệu. Ở đây chọn tôi chọn cấp độ hành chính để mô tả tính chất quan trọng của tài liệu. Ví dụ: với đề tài khoa học sẽ phân ra: cấp cơ sở, cấp tỉnh/thành phố, cấp bộ, cấp nhà nước, quốc tế. Ngoài ra một số tài liệu chỉ mang tính chất cá nhân.
- **`Tailieu`**: lưu trữ thông tin mô tả của tài liệu.
- **`User`**: lưu trữ thông tin người quản trị.

Lưu ý trong các cơ liệu dữ liệu này có bảng dữ liệu **`Tailieu`** có chứa đường link để chỉ đến tập tin nguồn là **`FileGoc`** và đường link đến tập tin đã trích xuất nội dung **`FileXML`**.

3.4. Rút trích dữ liệu

Vì các hệ thống phát hiện sao chép thường phải "băm" tài liệu vào những đơn vị có thể so sánh để xác định sự chồng chéo và phát hiện sao chép. Có nhiều phương pháp để chọn đơn vị so sánh như so sánh từ, câu, đoạn hoặc toàn bộ văn bản.

Vì vậy, khi xây dựng kho dữ liệu, chúng tôi trích dữ liệu từ các tài liệu gốc và tổ chức lưu trữ theo đơn vị nhỏ nhất là câu để phục vụ cho hệ thống phát hiện sao chép sử dụng làm đơn vị so sánh (nếu cần thì tách từ sau).

Để so sánh được nội dung hai văn bản với nhau, dữ liệu cần được lưu trữ ở dạng `Text` và qua quá trình xử lý như sau:

Tiền xử lý: làm sạch dữ liệu nhằm tối ưu hơn trong quá trình phát hiện sao chép. Chúng tôi lọc bỏ những phần nội dung không quan trọng như: lời mở đầu, lời cảm ơn, lời cam đoan, mục lục, hình ảnh, hình vẽ, công thức toán… Chỉ bóc lấy nội dung `Text` của tài liệu nhằm mục đích giảm thiểu thời gian so sánh tài liệu.

Tách đoạn: một tài liệu được xem như tập các đoạn. Khi tách đoạn chủ yếu dựa vào định dạng của văn bản để phát hiện biên giới của các đoạn. Đồng thời việc tách đoạn được thực hiện bằng cách quét qua các ký tự trong văn bản để tìm ra giới hạn của một đoạn. Giới hạn này là quy ước trong việc trình bày văn bản bao gồm dấu bắt đầu đoạn và dấu kết thúc đoạn. Trong quá trình tách đoạn, chúng tôi loại bỏ những đoạn có độ dài nhỏ như: tựa đề, tiêu đề,… nhằm tăng độ chính xác và giảm thời gian xử lý cho hệ thống phát hiện sao chép sau này.

Tách câu: ứng dụng xử lý ngôn ngữ tự nhiên và phân tích cú pháp để tách câu. Xác định ranh giới câu qua dấu câu như dấu chấm (.), dấu chấm than (!), dấu chấm hỏi (?) và dấu chấm phẩy (;).

Đánh chỉ mục: để lần vết trong quá trình tìm kiếm, xác định độ tương tự về sau, các đoạn và các câu trong từng bài báo cáo được đánh mã theo thứ tự.

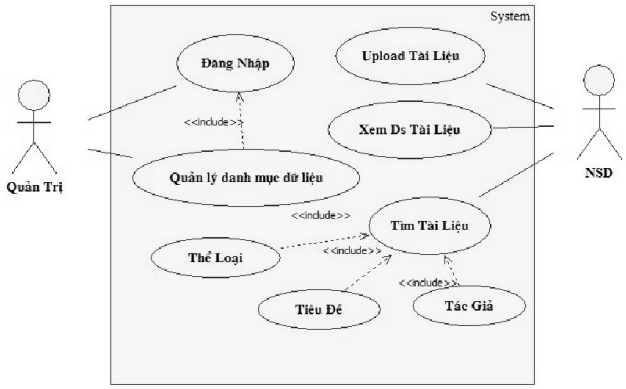
Sau khi xử lý dữ liệu chúng ta chuyển về định dạng XML để lưu vào kho. Bằng cách định nghĩa cấu trúc thích hợp, mỗi tài liệu được xử lý như trên và lưu trữ mỗi tài liệu dưới dạng một tập tin XML.

DTD của tập tin XML được đề xuất như sau:

```
<?xml version =”1.0” encoding=”UTF-8”?>
<Document>
   <chương id=”1”>
      <Paragraph id=”1”>
         <Sentence id=”1”>…..</Sentence>
         <Sentence id=”2”>…..</Sentence>
         ...
      </Paragraph>
      <Paragraph id=”2”>
         <Sentence id=”1”>…..</Sentence>
         <Sentence id=”2”>…..</Sentence>
         ...
      </Paragraph>
       ...
    </chuong>
    ...
</Document>
```

3.5. Quản lý nội dung kho dữ liệu

Để quản lý kho dữ liệu, chúng ta cần phải cập nhật, hiệu chỉnh, trích xuất,… thông tin trên kho dữ liệu này. Quan khảo sát, chúng tôi đề xuất biểu đồ ca sử dụng như sau:



Hình 4. *Biểu đồ ca sử dụng đối với kho dữ liệu*

Trong việc quản lý kho dữ liệu, `Quản trị` là cán bộ quản lý có thể quản lý danh mục và cấp quyền sử dụng cho các thành viên. `NSD` (người sử dụng như giảng viên, sinh viên, cán bộ của ĐHĐN) có thể cập nhật, xem danh sách các tài liệu và tìm kiếm tài liệu trong hệ thống.

4. Xây dựng kho dữ liệu

4.1. Lựa chọn công cụ phát triển

Để phát triển mã nguồn cho các chương trình, chúng tôi chọn sử dụng Visual Studio .NET. Đây là một môi trường tích hợp để phát triển phần mềm khá thuận tiện với Text Editor (hỗ trợ viết đoạn mã C#), Design View Editor (cài đặt giao diện người dùng và các điều khiển truy cập dữ liệu) và các chức năng hỗ trợ khác.

Về cơ sở dữ liệu, chúng tôi thực hiện trên SQL Server. Đây là hệ cơ sở dữ liệu mạnh mẽ, có khả năng đáp ứng được nhiều người sử dụng cùng lúc, có khả năng lưu trữ dữ liệu với dung lượng lớn và phân tán.

Ngoài ra, chúng tôi sử dụng chuẩn XML để lưu trữ dữ liệu đã rút trích để phục vụ việc phát hiện sao chép. XML là ngôn ngữ đánh dấu mở rộng được sử dụng rộng rãi, được nhiều ngôn ngữ, công cụ và nền tảng hỗ trợ. Đặc biệt, XML là tiêu chuẩn được sử dụng rộng rãi trên môi trường Internet.

4.2. Upload dữ liệu vào kho

Chúng tôi đã xây dựng hàm sao chép dữ liệu từ tập tin nguồn vào kho và nhập các thông tin mô tả vào CSDL.

Các bước upload tài liệu vào kho: kiểm tra tính đúng đắn của dữ liệu, nếu đúng chuyển sang bước 2, nếu sai yêu cầu nhập lại; Import thông tin vào CSDL, sao chép dữ liệu vào kho.

```
public class Uploader
{
  public static bool HasFile(String FieldName)
  {
   HttpPostedFile file =
   HttpContext.Current.Request.Files[FieldName];
   return(file != null && file.ContentLength>0);
   }
   public static String Save(String FieldName,
   String Folder, String FileName = null,
   String Ext = null)
   {
   HttpPostedFile file =
   HttpContext.Current.Request.Files[FieldName];
   string ext =
              Path.GetExtension(file.FileName);
   string orName =
    KhongDau(Path.GetFileNameWithoutExtension
    (file.FileName));
     int i = 1;
      if(FileName==null)
       {
        FileName = orName + ext;
       }
      else
       {
        orName = FileName;
        FileName = FileName + ext;
       }
   String path = Path.Combine(Folder, FileName);
     While
       (File.Exists(HttpContext.Current.Server
                                .MapPath(path)))
       {
         FileName = orName + i + ext;
         path = Path.Combine(Folder, FileName);
         i++;
       }
     file.SaveAs(HttpContext.Current.Server
                                .MapPath(path));
     return FileName;
}
```

4.3. Chuyển tập tin Word sang XML

Để tạo ra tập tin XML theo cấu trúc như đã đề xuất,

chúng tôi thực hiện các bước sau:

- **Bước 1:** Mở và duyệt tập tin Word, loại bỏ hình vẽ, hình ảnh và nội dung không quan trọng.
- **Bước 2:** Đếm số đoạn trong tập tin.
- **Bước 3:** Khởi tạo tài liệu XML rỗng.
- **Bước 4:** Xét duyệt từng đoạn và kiểm tra điều kiện, nếu đúng chuyển sang bước 5. Nếu điều kiện sai chuyển sang bước 6.
- **Bước 5:** Tạo node trong XML, tạo node cho phần tử đoạn, câu.
- **Bước 6:** Thoát và lưu tập tin XML.

4.4. Xây dựng giao diện Web cho hệ thống

Chúng tôi đã xây dựng một giao diện Web với chức năng như cập nhật dữ liệu, hiển thị danh sách tài liệu có trong kho và các thao tác xử lý khác. Tất cả các trang khác đều sử dụng trang này làm giao diện hiển thị những nội dung khác.

Dưới đây là giao diện chính của hệ thống:

Hình 5. *Giao diện hệ thống*

5. Kết luận

Chúng tôi đã tiến hành nghiên cứu việc triển khai hệ thống phát hiện nội dung sao chép trên một tài liệu từ các tài liệu sẵn có. Một trong những tiền đề quan trọng để triển khai hệ thống là phải có một kho dữ liệu lớn, có độ bao phủ cao đến lĩnh vực mà chúng ta muốn phát hiện việc sao chép. Trên cơ sở đó, chúng tôi đã triển khai bước đầu xây dựng hệ thống đó là xây dựng kho dữ liệu.

Chúng tôi đã hoàn thành việc thiết kế kho lưu trữ, xây dựng ứng dụng nhằm cập nhật, xử lý tự động dữ liệu trên kho như: cập nhật, sửa chữa, tìm kiếm, trích xuất nội dung,… Kho dữ liệu hiện nay đã có hơn 100 báo cáo luận văn/đồ án tốt nghiệp của sinh viên, hơn 100 báo cáo luận văn tốt nghiệp cao học, hơn 100 báo cáo tổng kết đề tài nghiên cứu khoa học và nhiều bài báo khoa học. Chúng ta có thể cập nhật thường xuyên dữ liệu vào kho với chức năng upload tài liệu dành cho người sử dụng.

Chúng tôi cũng đã xây dựng chương trình chuyển đổi tài liệu từ định dạng Word sang XML để phục vụ cho hệ thống phát hiện sao chép tài liệu.

Ngoài ra, hệ thống cũng đáp ứng được các yêu cầu phi chức năng như khả năng lưu trữ dữ liệu lớn, hệ thống chạy ổn định, giao diện đơn giản và dễ sử dụng.

Chúng tôi sẽ kiến nghị với ĐHĐN đưa ra yêu cầu bắt buộc sinh viên phải upload báo cáo luận văn hoặc đồ án tốt nghiệp lên kho dữ liệu trước khi nộp cho Hội đồng đánh giá. Nếu triển khai việc này, cơ sở dữ liệu sẽ được bổ sung một lượng dữ liệu lớn hàng năm và góp phần nâng cao chất lượng cho hệ thống phát hiện sao chép sau này.

Liên quan đến kho dữ liệu, chúng tôi sẽ tiếp tục nghiên cứu cách chuyển đổi tự động dữ liệu từ các định dạng khác như PDF, Latex, PPT, HTML,… sang tập tin XML. Ngoài ra, chúng tôi đang xây dựng mô-đun chương trình phát hiện sao chép dựa trên kho dữ liệu vừa thiết kế.

Nếu hệ thống này sớm được triển khai ứng dụng vào thực tế sẽ giúp cho việc hạn chế nạn "đạo văn" đang diễn ra khá phổ biến hiện nay, góp phần nâng cao chất lượng đào tạo và nghiên cứu khoa học tại ĐHĐN.

TÀI LIỆU THAM KHẢO

[1] S. Brin, J. Davis, H. Garcia-Molina, *Copy Detection Mechanisms for Digital Documents*, Proceedings of the ACM SIGMOD Annual Conference, San Francisco, CA, May 1995.

[2] N. Shivakumar and H. Garcia-Molina, *Building a Scalable and Accurate Copy Detection Mechanism*, Proceedings of 1st ACM International Conference on Digital Libraries (DL'96), March 1996, Bethesda Maryland.

[3] C. Xiao, W. Wang, X. Lin, J.X. Yu and G. Wang, *Efficient similarity joins for near-duplicate detection*, ACM Trans. Database Syst., Vol. 36. 10.1145/2000824.2000825, 2011.

[4] M. Potthast, A. Barron-Cedeno, B. Stein and P. Rosso, *Cross-language plagiarism detection*, Lang. Resour. Eval., 45: 45-62, 2011.

[5] N. Kang, A. Gelbukh, S.Y. Han, *Plagiarism Pattern Checker*, 2002.

[6] R.D. Smith, *Copy Detection Systems For Digital Documents*, Department of Computer Science, Master of Science, Brigham Young University, 1999.

[7] L. Guo, B. Jin and D. Huang, A Chunk-based Copy Detection Approach for Multimedia Documents, *Information Technology Journal,* 12: 2465-2469, 2013.

(BBT nhận bài: 05/11/2014, phản biện xong: 10/11/2014)