Kỷ yếu Hội nghị Quốc gia lần thứ X về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR), Đà Nẵng, ngày 17-18/08/2017
DOI: 10.15625/vap.2017.00057

MỘT CÁCH TIẾP CẬN MỚI ĐỂ PHÁT HIỆN SỰ GIỐNG NHAU CỦA VĂN BẢN DỰA TRÊN PHÉP BIẾN ĐỔI WAVELET RỜI RẠC

Hồ Phan Hiếu[1], Nguyễn Thị Ngọc Anh[1], Nguyễn Văn Hiếu[2], Đặng Thiên Bình[3], Võ Trung Hùng[1]

[1] Đại học Đà Nẵng, [2] Đại học Soongsil, [3] Đại học Sungkyunkwan

hophanhieu@ac.udn.vn, ntnanh@ued.udn.vn, hieuvnguyen@ssu.ac.kr, dtbinh@skku.edu, vthung@dut.udn.vn

TÓM TẮT: *Trong bài báo này, chúng tôi đề xuất một cách tiếp cận mới nhằm phát hiện sự giống nhau giữa các văn bản dựa trên phương pháp biến đổi Wavelet rời rạc (Discrete Wavelet Transform - DWT). Cụ thể là, các tài liệu gốc sẵn có được chuyển thành một tập các chuỗi số thực được gọi là các DNA (DeoxyriboNucleic Acid) nguồn thông qua DWT. Để kiểm tra sự giống nhau của một văn bản bất kỳ, chúng tôi cũng sử dụng DWT để tạo ra các DNA cho chính văn bản đó và tính toán khoảng cách Euclid nhỏ nhất từ các DNA này đến các DNA nguồn. Cuối cùng, bằng cách so sánh với một mức ngưỡng, các giá trị về khoảng cách sẽ cho biết đoạn văn bản được kiểm tra có giống với một văn bản nguồn nào đó hay không. Kết quả thực nghiệm chứng minh thuật toán do chúng tôi đề xuất đem lại hiệu quả cao trong phát hiện sự giống nhau của văn bản bằng cách thử nghiệm trên một bộ dữ liệu chuẩn tại Hội nghị quốc tế thường niên về phát hiện đạo văn (Plagiarism Analysis, Authorship Identification, and Near-Duplicate detection - PAN).*

Từ khóa: *Text Similarity, Discrete Wavelet Transformation, Text analysis, Data mining.*

I. GIỚI THIỆU

Với một lượng lớn các tài liệu số hóa được công bố công khai trên Internet hằng ngày thì việc phát hiện sao chép dựa trên nội dung văn bản là điều cần thiết. Trên thế giới, có nhiều kết quả nghiên cứu về so khớp văn bản phục vụ cho tìm kiếm, dịch tự động, trích chọn thông tin, tóm tắt văn bản, khai phá văn bản, web ngữ nghĩa, học máy,… có nhiều công trình nghiên cứu và nhiều ứng dụng hữu ích đối với tiếng Anh, trong đó có việc phát hiện sao chép [1-4]. Hiện nay, tuy có nhiều hệ thống phát hiện sao chép được ứng dụng vào thực tế nhưng vấn đề này vẫn còn nhiều thách thức như liên quan về kho dữ liệu, thuật toán, kỹ thuật, độ chính xác, độ tin cậy, hiệu năng, ngữ nghĩa, ngôn ngữ, trích dẫn, bản quyền tác giả,... đòi hỏi phải có nhiều nghiên cứu, hướng tiếp cận mới, giải pháp tối ưu hơn.

Phát hiện nội dung văn bản giống nhau là một bài toán khó, đang được nhiều nhóm nghiên cứu quan tâm. Các văn bản giống nhau hoàn toàn như bị sao chép toàn bộ, không có thay đổi nào thì dễ dàng phát hiện. Tuy nhiên, hầu hết là kiểu phát hiện các văn bản gần giống nhau, nên đây là một vấn đề khó hơn nhiều và các dạng gần giống là vô cùng đa dạng. Một văn bản có thể được sao chép toàn bộ hay chỉ một phần, các phần văn bản sao chép có thể bị thay đổi như thêm, sửa, xóa hoặc bị xáo trộn vị trí và nằm ở những vị trí bất kỳ của văn bản mới. Văn bản mới sau khi sao chép có thể chỉ sai khác với văn bản cũ ở một vài phần nhỏ hoặc cũng có thể không giống nhau bao nhiêu. Ngoài ra, còn có kiểu sao chép ý tưởng, chuyển ngữ,… Chính vì sự đa dạng trong việc sao chép văn bản mà không thể có một giải thuật hay kỹ thuật nào đo được một cách chính xác sự giống nhau giữa các văn bản. Bài toán này tuy không phải là mới, nhưng ở Việt Nam vẫn có rất ít những nghiên cứu tiền đề và ứng dụng rõ ràng được công bố. Phần sau sẽ trình bày các nghiên cứu liên quan về phương pháp biểu diễn và so khớp văn bản, đề xuất một cách tiếp cận mới và tóm tắt nội dung bài báo.

A. Các nghiên cứu liên quan

Sự tương đồng giữa hai văn bản là sự giống nhau về nội dung giữa hai văn bản đó. Do đó, hai văn bản là bản sao hoặc gần giống nhau thì sẽ có nội dung giống nhau nhiều, hay "độ tương đồng" giữa hai văn bản là cao. Độ tương đồng nằm trong khoảng giữa 0 và 1, nếu độ tương đồng càng gần 1 thì khả năng các văn bản gần giống nhau là cao và ngược lại [5]. Do đó, để kiểm tra các văn bản có giống nhau hay không ta phải tính độ tương đồng giữa chúng. Hiện có nhiều phương pháp để tính độ tương đồng văn bản như dựa trên so khớp chuỗi sử dụng các thuật toán Brute-Force, Morris-Pratt, Knuth-Morris-Pratt (KMP), Boyer-Moore, Karp-Rabin, Horspool,… [6]; sử dụng mô hình không gian vectơ để biểu diễn văn bản và dùng các độ đo khoảng cách để tính mức độ giống nhau như Euclid, Cosine, Jaccard, Dice,… [7]; các phương pháp dựa trên từ vựng, dựa trên cơ sở dữ liệu, dựa trên tri thức [8]; các nghiên cứu về xử lý ngôn ngữ tự nhiên… Mỗi thuật toán, phương pháp so khớp có một hướng tiếp cận khác nhau và mỗi thuật toán đều có những ưu điểm và hạn chế riêng.

Một trong những nhiệm vụ đầu tiên trong việc xử lý văn bản là lựa chọn mô hình biểu diễn văn bản thích hợp để đem lại hiệu quả trong tính toán, xử lý. Một văn bản ở dạng thô (dạng chuỗi ký tự) cần được chuyển sang một mô hình khác để tạo thuận lợi cho việc biểu diễn và tính toán, tùy thuộc vào từng thuật toán xử lý khác nhau mà chọn mô hình biểu diễn riêng. Trong các bài toán xử lý văn bản, mô hình không gian vectơ được sử dụng phổ biến nhất, mô hình này biểu diễn văn bản như một vectơ đặc trưng của các thuật ngữ/từ xuất hiện trong toàn bộ tập văn bản, trọng số các đặc trưng thường được tính thông qua độ đo TF-IDF [9].

Qua khảo sát, chúng tôi nhận thấy hầu hết các phương pháp sử dụng để phát hiện sao chép thường dựa trên mô hình không gian vectơ để biểu diễn văn bản và sử dụng các độ đo khoảng cách giữa các vectơ để tính toán mức độ giống nhau. Nhìn chung, các nghiên cứu tập trung vào so khớp từ sử dụng phương pháp n-gram, dấu vân tay, thống kê tần suất.

B. Đề xuất và đóng góp của bài báo

Phép biến đổi Wavelet rời rạc (Discrete Wavelet Transform - DWT) được sử dụng chủ yếu trong xử lý tín hiệu số như mã hóa tiếng nói, xử lý ảnh, ứng dụng trong lọc nhiễu, nhận dạng và trong các lĩnh vực khác như công nghiệp, điện tử, viễn thông, y học,… [10]. Tuy nhiên, hầu như chưa có công trình nghiên cứu về DWT được ứng dụng trong lĩnh vực xử lý văn bản.

Trong nghiên cứu này, chúng tôi đề xuất một cách tiếp cận hoàn toàn mới trong phát hiện sự giống nhau của văn bản, đó là dựa trên phương pháp DWT. Phương pháp này được chúng tôi đề xuất và thực hiện qua các bước chính gồm: (1) Các tài liệu gốc sẵn có được chuyển thành một tập các chuỗi số thực được gọi là các DNA nguồn thông qua DWT; (2) Để kiểm tra sự giống nhau của một văn bản bất kỳ, chúng tôi cũng sử dụng DWT để tạo ra các DNA cho chính văn bản đó và tính toán khoảng cách Euclid nhỏ nhất từ các DNA này đến các DNA nguồn; (3) So sánh với một mức ngưỡng, các giá trị về khoảng cách sẽ cho biết đoạn văn bản được kiểm tra có giống với văn bản nguồn nào đó hay không. Ưu điểm của việc chuyển đổi văn bản sang chuỗi số thực theo phương pháp đề xuất này làm cho việc lưu trữ rất thuận tiện và giảm độ phức tạp tính toán hơn nhiều so với sử dụng văn bản chứa chuỗi ký tự. Thêm vào đó, cách mã hoá như đề xuất hoàn toàn giữ nguyên được thông tin của từ được mã hoá.

Các kết quả thực nghiệm chứng minh thuật toán do chúng tôi đề xuất đem lại hiệu quả cao trong phát hiện sự giống nhau của văn bản bằng cách thử nghiệm trên một bộ dữ liệu huấn luyện chuẩn của PAN và đạt được độ chính xác trên 97%.

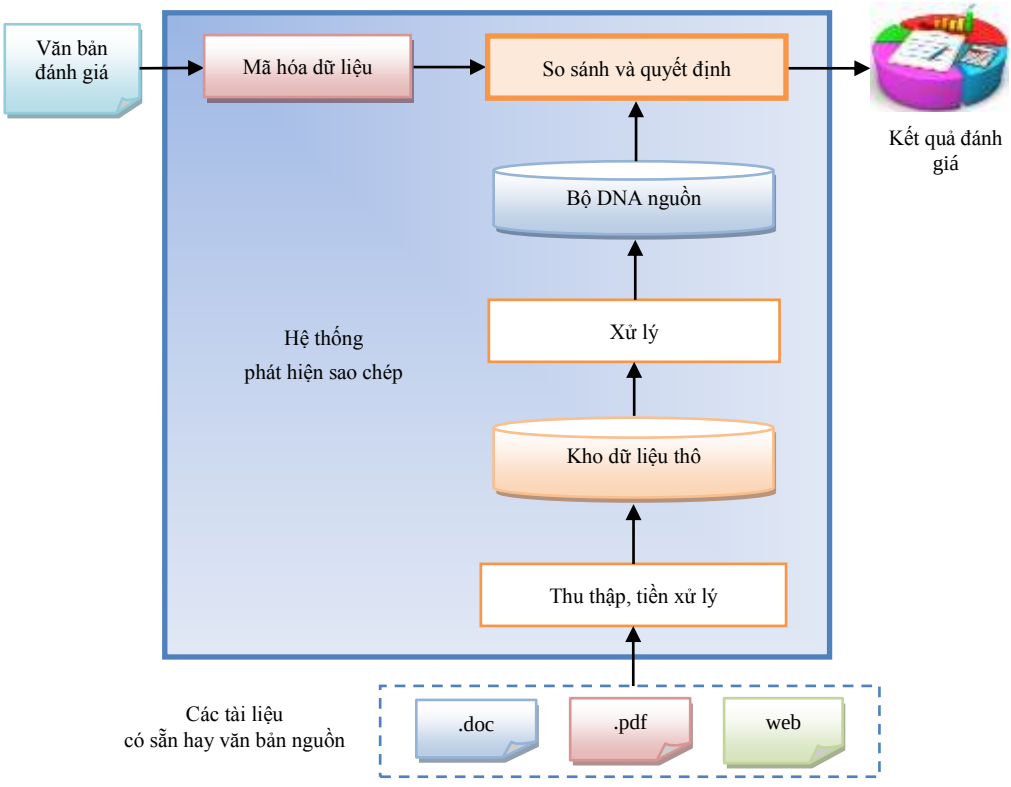
C. Tóm tắt nội dung bài báo

Nội dung còn lại của bài báo này được tổ chức thành bốn phần. Phần II mô tả tổng quan về hệ thống đề xuất và cách tìm DNA sử dụng bộ lọc Haar. Trong III, chúng tôi sẽ phân tích giải thuật đề xuất liên quan đến tiền xử lý dữ liệu, xây dựng bộ DNA cho văn bản gốc và mô tả thuật toán phát hiện sự giống nhau. Phần IV trình bày kết quả thực nghiệm trên bộ dữ liệu chuẩn của PAN và kiểm tra giải thuật đề xuất. Cuối cùng, chúng tôi sẽ đưa ra kết luận và hướng phát triển của bài báo trong tương lai.

II. ĐỀ XUẤT HỆ THỐNG PHÁT HIỆN SAO CHÉP VĂN BẢN

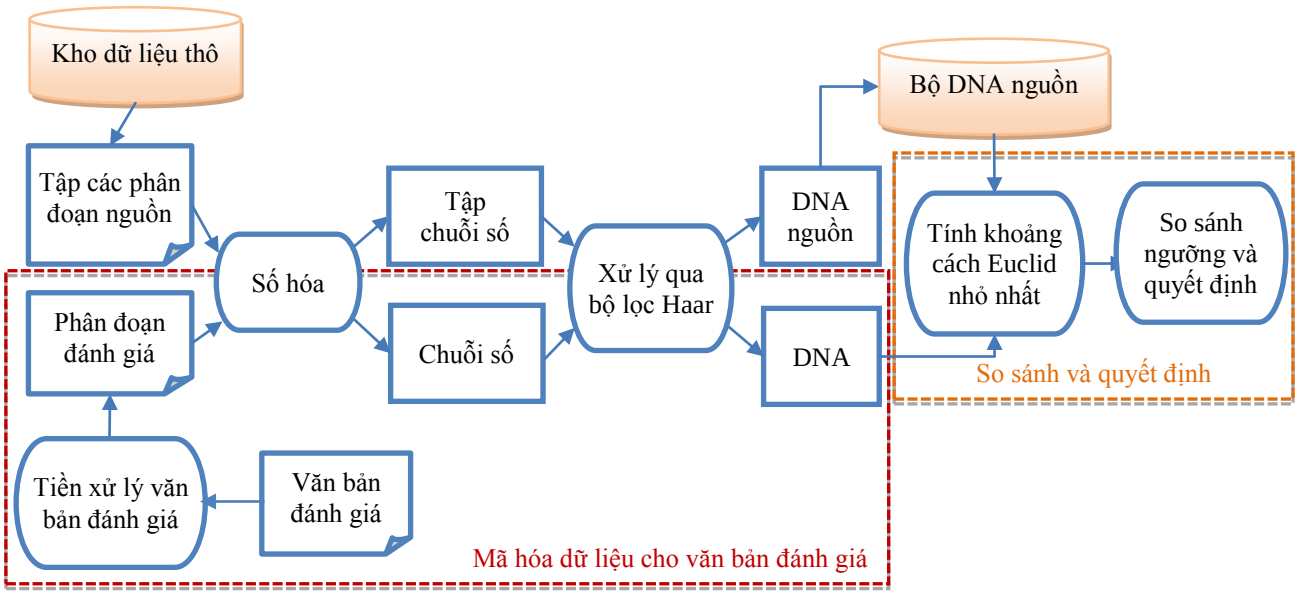
A. Đề xuất hệ thống và quy trình xử lý

Với mục tiêu cuối cùng là để nghiên cứu, xây dựng một hệ thống phát hiện sao chép văn bản, chúng tôi đã nghiên cứu, khảo sát các tài liệu tổng quan về hệ thống phát hiện sao chép văn bản [1-4], cũng như những nghiên cứu về DWT, bộ lọc Haar [11]. Trên cơ sở đó, chúng tôi đề xuất mô hình tổng quan của hệ thống như Hình 1.



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

Trong bài báo này, chúng tôi tập trung vào việc thiết kế các khối cho hệ thống phát hiện sao chép văn bản. Trước tiên, các tài liệu sẵn có được thu thập lại, đồng thời quá trình tiền xử lý sẽ loại bỏ các dấu câu, ký tự đặc biệt và lưu trữ dưới dạng dữ liệu thô. Để thuận tiện cho quá trình xử lý chính, trong giai đoạn tiền xử lý, văn bản thu thập sẽ được phân đoạn và lấy mẫu sao cho các mẫu có độ dài bằng nhau. Sau đó, các phân đoạn này được lưu trữ như là dữ liệu thô nhằm mục đích trích xuất các đoạn văn bản giống nhau (nếu có) tại đầu ra kết quả đánh giá. Trong giai đoạn xử lý chính, các văn bản sẽ được số hóa và cho qua bộ lọc Haar để thu được dữ liệu cho bộ DNA nguồn. Trong khi đó, văn bản đánh giá được cho qua bộ mã hóa để xử lý. Bộ mã hóa này thực chất giống như quá trình tính toán DNA cho văn bản nguồn. Tuy nhiên, văn bản đánh giá thô được tạo thành sau quá trình tiền xử lý sẽ được phân đoạn. Sau đó, từng phân đoạn trong văn bản đánh giá được mã hóa thành một DNA nhằm mục đích phát hiện sự giống nhau (nếu có) của phân đoạn đó với một phân đoạn khác thuộc bộ dữ liệu nguồn. Vì vậy, văn bản đánh giá thô có thể xem như các chuỗi tín hiệu cần xử lý. Hình 2 mô tả chi tiết quá trình xử lý để đánh giá văn bản kiểm tra so với tập văn bản nguồn (kho dữ liệu).



Hình 2. Sơ đồ mô tả chi tiết toàn bộ quá trình xử lý để đánh giá văn bản kiểm tra so với tập văn bản nguồn

B. Cơ sở lý thuyết và giải thuật cho bộ lọc Haar

Có thể nhận thấy rằng bộ lọc Haar dùng cho việc tính toán các DNA đóng vai trò rất quan trọng cho việc so sánh và đưa ra quyết định. Việc xử lý qua bộ lọc Haar được sử dụng nhiều lần cho cả văn bản nguồn và văn bản cần đánh giá. Vì vậy, trong phần này chúng tôi giới thiệu cơ sở lý thuyết và phát triển giải thuật tạo ra các chuỗi DNA cho bộ lọc Haar, sau đó các thuật toán chính sẽ dùng lặp lại giải thuật này như một module chuẩn.

Trong DWT, đường Haar Wavalet hay được gọi là bộ lọc Haar được sử dụng phổ biến trong khai phá dữ liệu chuỗi thời gian và lập chỉ mục [12]. Qua nghiên cứu về DWT và đường Haar, để so sánh mức độ giống nhau giữa hai chuỗi, chúng tôi đề xuất ý tưởng nhằm chuyển đổi nội dung văn bản thành dạng chuỗi thời gian thực (thông qua bộ số hóa) và sử dụng bộ lọc Haar trong DWT để phát hiện ra các mẫu bất thường, dữ liệu văn bản được chuyển đổi và biểu diễn thành những dãy số thực được biểu diễn bởi $\mathbf{x} = [x_1\ x_2\ \ldots\ x_N]$. Trong phần này, chúng tôi chỉ phân tích cơ sở lý thuyết và phát triển giải thuật cho bộ lọc Haar nhằm tạo ra các DNA được sử dụng trong hệ thống xử lý phát hiện giống nhau. Ở phần sau, chúng tôi sẽ phân tích sâu hơn các khối tiền xử lý và khối xử lý gồm số hóa, tổ chức bộ DNA và xác định khoảng cách Euclid nhỏ nhất.

Theo như Hình 2, tín hiệu vào bộ lọc là các chuỗi số rời rạc gồm $N = 2^K$ số thực. Một phép biến đổi Haar rời rạc được thực hiện qua K bước lặp và tại lần lặp thứ k (hay mức thứ k), với $k = 1,2,\ldots,K$; tín hiệu đầu ra của phép biến đổi được mô tả như sau:

$$\mathbf{x}^{(k)} = \left[\mathbf{x}_{\text{low}}^{(k)}\ \mathbf{x}_{\text{high}}^{(k)}\ \mathbf{x}_{\text{c}}^{(k-1)}\right] \tag{1}$$

trong đó, các hệ số xấp xỉ $\mathbf{x}_{\text{low}}^{(k)}$ và hệ số chi tiết $\mathbf{x}_{\text{high}}^{(k)}$ được cho bởi công thức lấy mẫu con như sau:

$$\mathbf{x}_{\text{low}}^{(k)} = \left(\mathbf{x}_{\text{a}}^{(k-1)} * \mathbf{f}_{\text{L}}\right) \downarrow 2, \tag{2}$$

$$\mathbf{x}_{\text{high}}^{(k)} = \left(\mathbf{x}_{\text{a}}^{(k-1)} * \mathbf{f}_{\text{H}}\right) \downarrow 2, \tag{3}$$

với $\mathbf{f}_{\text{L}} = [1\ 1]$ và $\mathbf{f}_{\text{H}} = [-1\ 1]$ lần lượt là đáp ứng bộ lọc thông thấp và thông cao; $\mathbf{x}_{\text{a}}^{(k-1)}$ và $\mathbf{x}_{\text{c}}^{(k-1)}$ tương ứng là các hệ số xấp xỉ của bước thứ (*k-1)* và tổng hợp các hệ số chi tiết của chuỗi tín hiệu thu được tại các bước trước đó.

Tại điểm khởi tạo, $\mathbf{x}_{\text{a}}^{(0)}$ và $\mathbf{x}_{\text{c}}^{(0)}$ được cho như sau:

$$\mathbf{x}_{\text{a}}^{(0)} = \mathbf{x}^{(0)}, \tag{4}$$

$$\mathbf{x}_{\text{c}}^{(0)} = [], \tag{5}$$

trong đó, $\mathbf{x}^{(0)}$ là chuỗi tín hiệu ban đầu (vectơ x sau khi được số hóa) và [] là vectơ rỗng. Các giá trị $\mathbf{x}_{\text{a}}^{(\text{k})} \in \mathbb{R}^{1\times N_{\text{a}}(k)}$, với $N_{\text{a}}(k) = 2^{K-k}$, $\mathbf{x}_{\text{c}}^{(k)} \in \mathbb{R}^{1\times N_{\text{c}}(k)}$ và $N_{\text{c}}(k) = \sum_{i=1}^{k} 2^{K-i}$, *k = 1,2,…,K* sẽ được cập nhật theo công thức sau:

$$\mathbf{x}_{\text{a}}^{(k)} = \mathbf{x}_{\text{low}}^{(k)}, \tag{6}$$

$$\mathbf{x}_{\text{c}}^{(k)} = \left[\mathbf{x}_{\text{high}}^{(k)}\ \mathbf{x}_{\text{c}}^{(k-1)}\right]. \tag{7}$$

Có thể dễ dàng chứng minh $N_{\text{a}}(k) + N_{\text{c}}(k) = 2^{K-k} + \sum_{i=1}^{k} 2^{K-i} = 2^{K} = N$, *k = 1,2,…,K*. Qua đó, tín hiệu sau *K* bước lặp vẫn có độ dài *N* như ban đầu. Thực chất, các phân đoạn văn bản khác nhau sau khi được số hóa và qua bộ lọc Haar sẽ cho ra các chuỗi số mang thông tin đặc trưng có thể phân biệt được mức độ khác nhau giữa chúng. Do đó, các chuỗi tín hiệu sau bộ lọc được gọi là các DNA.

Cuối cùng chúng tôi phát triển một thuật toán theo phân tích ở trên như sau:

Bảng 1. Thuật toán xác định giá trị cho các chuỗi DNA

	Thuật toán 1: Xác định các chuỗi DNA
1:	**Đầu vào:** Tập chuỗi số thực.
2:	**Khởi tạo:** Các vectơ xấp xỉ và hệ số được cho bởi công thức (4) và (5)
3:	**For** *k*:= 1→ *K*
4:	Tính chuỗi số thứ *k* theo công thức (1), (2) và (3)
5:	Cập nhật giá trị cho vectơ xấp xỉ và hệ số theo công thức (6) và (7)
6:	**End**
7:	**Đầu ra:** Chuỗi số thứ *K* chính là DNA cần tính.

III. THUẬT TOÁN ĐỀ XUẤT

Trong phần này, chúng tôi sẽ phân tích chi tiết nhiệm vụ của từng khối trong quá trình xử lý. Cụ thể là, giai đoạn tiền xử lý sẽ loại bỏ các ký tự đặc biệt và chia văn bản thành các phân đoạn. Tiếp đến, khâu số hóa ở giai đoạn xử lý chính sẽ chuyển đổi các từ trong từng phân đoạn thành một số thực đặc trưng trước khi đưa đến bộ lọc Haar để lấy mẫu cho việc tính toán các DNA nhằm phục vụ cho việc so sánh và quyết định ở khối sau. Để thuận tiện cho việc phân tích chi tiết các quy trình, chúng tôi sẽ trình bày phần tiền xử lý dữ liệu chung cho cả tập văn bản nguồn và văn bản đánh giá, sau đó phần xử lý chính sẽ được mô tả riêng cho tập văn bản nguồn và văn bản đánh giá.

A. Tiền xử lý dữ liệu

Các văn bản nguồn sau khi được thu thập sẽ được đưa đến bộ tiền xử lý dữ liệu. Tại đây, hệ thống sẽ xem xét một câu như là đơn vị nhỏ nhất có thể bị sao chép. Theo đó, các phân đoạn sẽ được xác định dựa trên các dấu chấm câu (.). Sau đó, các ký tự không phải chữ số và chữ cái alphabet như (! ? , []…) sẽ bị loại bỏ khỏi các phân đoạn. Nói cách khác, một phân đoạn sẽ đại diện cho một câu trong văn bản sau khi được loại bỏ các ký tự đặc biệt. Các phân đoạn này sẽ được lưu trữ như văn bản thô cho mục đích truy xuất kết quả.

Văn bản đánh giá cũng được tiền xử lý một cách tương tự và cho một văn bản đánh giá thô. Văn bản đánh giá thô này có thể được hệ thống cập nhật như dữ liệu mới nhằm mục đích mở rộng cơ sở dữ liệu nguồn cho việc đánh giá các văn bản khác.

B. Xây dựng bộ DNA cho văn bản nguồn

Để xây dựng bộ DNA cho văn bản nguồn, chúng tôi thực hiện việc số hóa các từ trong mỗi phân đoạn. Tuy nhiên, có thể nhận thấy rằng mỗi phân đoạn thu thập được từ các văn bản nguồn là một câu. Theo đó, số lượng từ trong mỗi phân đoạn sẽ khác nhau, điều này dẫn đến độ dài chuỗi tín hiệu số thực cho mỗi phân đoạn là không giống nhau. Trong khi đó, việc tính toán các DNA sử dụng bộ lọc Haar (như mục IIB) đòi hỏi độ dài các chuỗi tín hiệu phải bằng nhau. Vì vậy, chúng tôi sử dụng một cửa sổ để lấy mẫu cho mỗi phân đoạn. Cửa sổ lấy mẫu này sẽ dịch trên mỗi phân đoạn để trích ra các mẫu theo một tỉ lệ xếp chồng lên mẫu trước đó được cho sẵn. Để giảm thiểu số lần số hóa các từ trong phân đoạn, chúng tôi sẽ thực hiện việc số hóa trên toàn phân đoạn để tạo ra một chuỗi số tín hiệu thực trước khi lấy mẫu dữ liệu. Phần sau là nội dung chi tiết của việc số hóa và chuẩn hóa dữ liệu, lấy mẫu cho từng phân đoạn, tổ chức dữ liệu cho bộ DNA nguồn.

1. Số hóa và chuẩn hóa dữ liệu

Ở bước này, chúng tôi sẽ số hóa mỗi từ trong phân đoạn thành một số thực mang đặc trưng của từ đó và phân biệt với số thực đại diện cho từ khác. Để làm được điều này, trước hết chúng tôi sử dụng bảng mã Unicode để chuyển đổi các ký tự trong mỗi từ thành một số nguyên ở hệ thập phân, sau đó ghép chuỗi các số nguyên này theo đúng trật tự tương ứng của chúng trong một từ để tạo thành một số nguyên đại diện. Tuy nhiên, số lượng chữ số cho mỗi ký tự trong bảng mã là khác nhau, nên sẽ xảy ra trường hợp một chuỗi số đã ghép có thể được tạo thành bởi nhiều từ khác nhau. Do đó, chúng tôi chuẩn hóa giá trị cho mỗi ký tự bằng cách xác định một số lượng chữ số tối đa cho mỗi ký tự dựa vào bảng mã Unicode, số lượng chữ số tối đa này gọi là *m*. Ví dụ, nếu một ký tự có giá trị trong bảng mã Unicode gồm *m'* chữ số, với *m'*<*m*, giá trị đó sẽ được thêm *m''* số 0 vào trước sao cho *m'*+*m''* = *m*. Điểm nổi bật của cách chuẩn hóa này là vị trí của giá trị số hóa cho mỗi ký tự sau khi ghép chuỗi sẽ được duy trì tương ứng với vị trí của nó trong một từ. Giả sử một từ thứ *i* nào đó trong phân đoạn có L_i ký tự, trong đó ký tự thứ *l* (l = 1,2,...,L_i) có giá trị theo bảng mã Unicode là $s_{i,l}$. Số nguyên đại diện của từ đó trong một phân đoạn (ký hiệu là s_i) sẽ được tính theo công thức sau:

$$s_i = \sum_{l=1}^{L_i} s_{i,l} \times 10^{m\times\left(L_i - l\right)} \tag{8}$$

Tuy nhiên, mỗi từ trong phân đoạn có độ dài khác nhau nên các số nguyên đại diện cho một từ sẽ có độ lớn rất khác nhau. Để việc so sánh được chính xác, chúng tôi chuẩn hóa giá trị cho một từ theo hàm logaric cơ số 10. Giả sử độ dài lớn nhất của chuỗi số mà một từ có khả năng tạo thành là $M = m*L_{max}$, với L_{max} là độ dài từ tối đa có thể. Cuối cùng, giá trị số thực đại diện cho một từ được sử dụng như tín hiệu được đưa đến bộ lọc Haar ở khối tiếp theo được xác định bởi công thức sau:

$$x_i^{\mathrm{w}} = M - m\times L_i + \log_{10}\left(s_i\right) \tag{9}$$

2. Lấy mẫu cho từng phân đoạn

Giả sử một phân đoạn có *W* từ và từ thứ *i* (*i* = 1,2,…,*W)* được đặc trưng bởi giá trị x_i^{w} như công thức (9). Để thực hiện việc tạo DNA cho phân đoạn, chúng tôi thực hiện việc lấy mẫu cho phân đoạn đó. Cụ thể là, một cửa sổ lấy mẫu độ dài là $N = 2^K$ tín hiệu được sử dụng để trích ra *N* giá trị số thực liên tiếp trong *W* giá trị của phân đoạn bằng cách dịch từ trái qua phải. Do đó, một phân đoạn có thể tạo ra nhiều mẫu và các mẫu đều có độ dài bằng nhau là *N*. Độ dịch của cửa sổ có thể là một hoặc nhiều giá trị. Khoảng dịch càng xa thì độ phức tạp tính toán sẽ được giảm nhưng đồng thời độ chính xác cũng giảm theo. Tóm lại, đầu vào của bộ lọc Haar chính là một mẫu trong phân đoạn được cho bởi vectơ sau:

$$\mathbf{x} = \mathbf{x}^{(0)} = \left[x_1\ x_2 \ldots x_N\right] = \left[x_i^{\mathrm{w}} x_{i+1}^{\mathrm{w}} \ldots x_{i+N-1}^{\mathrm{w}}\right] \tag{10}$$

Các mẫu này sau đó sẽ được đưa đến bộ lọc Haar để tạo thành tập các DNA của một phân đoạn.

3. Tổ chức dữ liệu cho bộ DNA nguồn

Sau khi thực hiện các bước như ở mục 1 và 2, chúng ta sẽ có được một bộ DNA cho tập các văn bản thu thập được. Chúng tôi sắp xếp bộ DNA theo giá trị đầu tiên của DNA tăng dần. Mục đích của việc sắp xếp là để hệ thống có thể thực hiện việc tìm kiếm nhị phân để xác định DNA giống với DNA của một mẫu thuộc phân đoạn nào đó trong văn bản đánh giá. Qua đó, chúng tôi có thể cải thiện được độ phức tạp của thuật toán đánh giá văn bản. Sở dĩ có thể dùng giá trị đầu tiên của DNA làm khóa sắp xếp vì đó chính là giá trị xấp xỉ hay tổng của các giá trị thành phần sau *K* bước lặp. Vì vậy, tại vị trí này nếu giá trị của hai mẫu DNA (một mẫu thuộc văn bản nguồn và một mẫu văn bản đánh giá)

giống nhau, hai mẫu văn bản tương ứng với hai DNA này sẽ giống nhau. Tuy nhiên, chúng ta cần thêm một khâu so sánh ngưỡng trước khi đưa ra quyết định, khâu này sẽ được đề cập chi tiết ở phần sau.

Bảng 2. Thuật toán lưu trữ bộ DNA nguồn

Thuật toán 2: Tính toán bộ DNA nguồn	
1:	**Đầu vào:** Tập các văn bản nguồn thu thập được.
2:	**Khởi tạo:** Độ dài cho một DNA (*N*).
3:	Tiền xử lý, phân đoạn và lưu trữ văn bản nguồn.
4:	**For** mỗi phân đoạn, cần thực hiện:
5:	Số hóa phân đoạn.
6:	Lấy mẫu và xác định nhóm DNA của phân đoạn theo thuật toán 1.
7:	**For** mỗi DNA trong nhóm, cần thực hiện:
8:	Tìm kiếm nhị phân trên kho dữ liệu để xác định vị trí cần lưu trữ của các DNA sao cho các giá trị đầu tiên của các chuỗi DNA trong toàn bộ kho dữ liệu được sắp xếp tăng dần. Chèn các DNA vào kho dữ liệu theo đúng vị trí đã tìm.
9:	**EndFor** // *kết thúc vòng lặp for dòng 7*
10:	**EndFor** // *kết thúc vòng lặp for dòng 4*
11:	**Kết quả thu được:** Kho dữ liệu đã được cập nhật và sắp xếp.
12:	

C. Mô tả thuật toán phát hiện sự giống nhau

1. Mã hóa dữ liệu và tính DNA của văn bản đánh giá

Sau khi tiền xử lý văn bản đánh giá, chúng ta có thể dễ dàng thực hiện quy trình mã hóa dữ liệu văn bản đánh giá như mục B1 và B2. Đối với tập văn bản nguồn, các DNA của chúng được lưu trữ như cơ sơ dữ liệu, giống như thư viện sẵn có để đối chiếu sự giống nhau. Trong khi đó, văn bản đánh giá sau khi được phân đoạn sẽ được đưa vào mã hóa tuần tự như tín hiệu chuỗi thời gian thực. Theo đó, mỗi phân đoạn sẽ được lấy mẫu thành một nhóm các DNA và nhóm các DNA này sẽ được chuyển đến bộ so sánh. Sau đó, nhóm các DNA của phân đoạn sau cũng theo một quy trình như vậy cho đến khi hết văn bản đánh giá.

2. So sánh và đưa ra quyết định

Ở khối cuối cùng của hệ thống, chúng tôi sẽ so sánh từng nhóm DNA của các phân đoạn với các DNA của tập dữ liệu nguồn được lưu trữ sẵn. Đối với mỗi mẫu DNA trong nhóm DNA đưa vào khâu so sánh, chúng tôi sẽ tìm kiếm nhị phân trong kho dữ liệu để xác định DNA nguồn nào có giá trị đầu tiên giống với DNA đang xét nhất. Tiếp theo, khoảng cách Euclid giữa hai DNA được tính rất đơn giản theo công thức sau:

$$d\left(\mathbf{x},\mathbf{y}\right)=\left\|\mathbf{x}-\mathbf{y}\right\|_2^2 \tag{11}$$

trong đó, $\mathbf{x}\in\mathbb{R}^{1\times N}$ và $\mathbf{y}\in\mathbb{R}^{1\times N}$ lần lượt là vectơ DNA nguồn và vectơ DNA đang xét. Khoảng cách Euclid này sẽ được so sánh với một mức ngưỡng ε. Nếu $d(\mathbf{x}, \mathbf{y})< \varepsilon$, hai DNA được xem là giống nhau và vị trí tương ứng với DNA đang xét được đánh dấu lại để hệ thống đưa ra quyết định sau khi tổng hợp tất cả các mẫu DNA của phân đoạn. Thuật toán phát hiện giống nhau giữa hai văn bản được mô tả trong Bảng 3.

Bảng 3. Thuật toán phát hiện sự giống nhau giữa hai văn bản

Thuật toán 3: Phát hiện sự giống nhau giữa hai văn bản	
1:	**Đầu vào:** Văn bản cần đánh giá.
2:	**Khởi tạo:** Độ dài chuỗi DNA (*N*) và mức ngưỡng (ε) so sánh độ giống nhau.
3:	Tiền xử lý, phân đoạn và lưu trữ dữ liệu để xuất kết quả.
4:	**For** mỗi phân đoạn, cần thực hiện:
5:	Số hóa phân đoạn.
6:	Lấy mẫu và xác định nhóm DNA của phân đoạn theo Thuật toán 1.
7:	**For** mỗi DNA **y** nào đó trong nhóm, cần thực hiện:
8:	Tìm kiếm nhị phân trên kho DNA nguồn để tìm ra một DNA **x** sao cho giá trị đầu của chuỗi DNA **y** đang xét gần nhất với giá trị đầu của DNA **x**.
9:	Tính khoảng cách Euclid d(x, y) theo công thức (11).
10:	**If** $d(\mathbf{x}, \mathbf{y})< \varepsilon$ **then**
11:	Đánh dấu DNA **y** đang xét.
12:	**Endif**
13:	**Endfor** // *kết thúc vòng lặp for dòng 7*
14:	Tổng hợp các DNA **y** đã đánh dấu (nếu có) để thu được chuỗi các từ giống nhau của phân đoạn đang xét so với phân đoạn nguồn.
15:	**Endfor** // *kết thúc vòng lặp for dòng 4*
16:	**Kết quả thu được:** Đưa ra các phân đoạn chứa các từ giống với các phân đoạn nguồn (nếu có).

IV. KẾT QUẢ THỰC NGHIỆM

Qua các bước xử lý chính như trên, chúng tôi đã chuyển đổi được văn bản thành dạng tín hiệu số là các chuỗi số thực DNA, đảm bảo được tính duy nhất và toàn vẹn của thông tin. Với phương pháp đề xuất này, chúng tôi đã xây dựng các module xử lý, trong đó có mã hóa các văn bản nguồn thành bộ DNA nguồn và tổ chức lưu trữ xếp hàng theo các lớp thông qua các giá trị khóa nên tốc độ tính toán của hệ thống nhanh và độ chính xác rất cao.

Để kiểm tra kết quả của giải thuật đề xuất, chúng tôi sử dụng các phép đo trong PAN [13] để tính các giá trị *prec* (precision) và *rec* (recall). Một cách cụ thể, chúng tôi gọi tập các chuỗi ký tự bị sao chép và tập chuỗi ký tự được phát hiện lần lượt như sau:

$$\mathcal{S} = \{S\} \tag{12}$$

$$\mathcal{D} = \{D\} \tag{13}$$

trong đó, $\mathcal{S}$ và $\mathcal{D}$ lần lượt là các chuỗi văn bản nguồn bị sao chép và các chuỗi văn bản đánh giá được phát hiện là giống với các chuỗi trong văn bản nguồn; với việc tính $\mathcal{S}$ và $\mathcal{D}$ nhằm đưa ra tỉ lệ giống nhau của văn bản đánh giá so với các văn bản nguồn bị sao chép. Các giá trị *prec* và *rec* được xác định bởi các công thức theo [13], đó là:

$$prec = \frac{1}{|\mathcal{D}|} \sum_{D \in \mathcal{D}} \frac{\left| D \cap \left(\bigcup_{S \in \mathcal{S}} S \right) \right|}{|D|} \tag{14}$$

$$rec = \frac{1}{|\mathcal{S}|} \sum_{S \in \mathcal{S}} \frac{\left| S \cap \left(\bigcup_{D \in \mathcal{D}} D \right) \right|}{|S|} \tag{15}$$

trong đó, $|\mathcal{S}|$ và $|\mathcal{D}|$ lần lượt là số phần tử trong tập hợp $\mathcal{S}$ và $\mathcal{D}$, $|\mathcal{S}|$ và $|\mathcal{D}|$ lần lượt là độ dài của chuỗi $S \in \mathcal{S}$ và $D \in \mathcal{D}$.

Qua 10 lần thử nghiệm trên bộ dữ liệu huấn luyện của PAN năm 2009 [14], mỗi lần đánh giá 100 văn bản nghi ngờ hoàn toàn khác với văn bản sử dụng để tìm giá trị ngưỡng ε. Chúng tôi thiết lập các giá trị như Bảng 4 và kết quả đạt được như Bảng 5 và Hình 3.

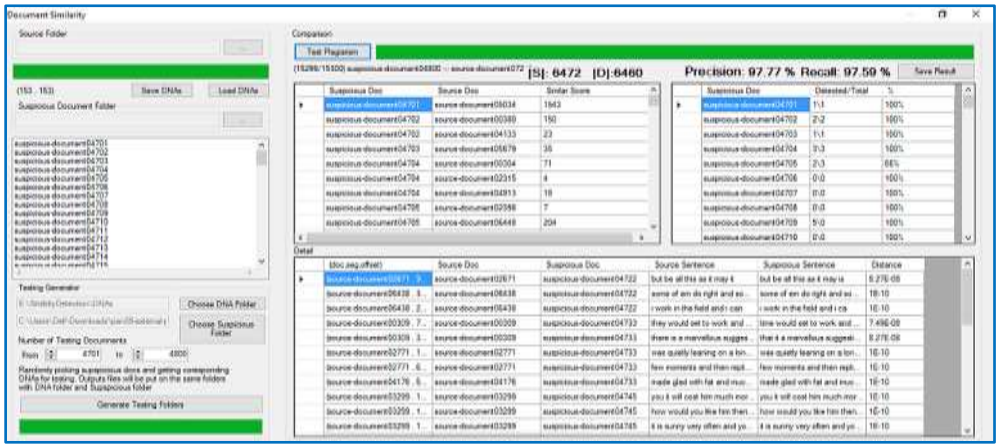
Bảng 4. Các giá trị thiết lập cho quá trình thử nghiệm

Thông số	Giá trị thiết lập
Số lượng chữ số tối đa mã hóa một ký tự, *m*	5
Số ký tự tối đa trong một từ, L_{max}	45
Độ dài mẫu DNA, *N*	8
Số bước lặp để xử lý với bộ lọc Haar, *K*	3
Giá trị ngưỡng cho khoảng cách Euclid, ε	10^{-7}

Bảng 5. Kết quả thực nghiệm

Lần thử nghiệm	$\lvert\mathcal{S}\rvert$	$\lvert\mathcal{D}\rvert$	*prec* (%)	*rec* (%)
1	6028	6004	98.12	97.73
2	5336	5315	97.99	97.60
3	9340	9310	98.02	97.71
4	6491	6455	97.86	97.32
5	8063	8033	97.77	97.41
6	6982	6960	97.63	97.32
7	6472	6460	97.77	97.59
8	5621	5595	98.14	97.69
9	6519	6506	97.86	97.67
10	6026	6005	98.00	97.66
Giá trị trung bình	***6687.8***	***6664.3***	***97.92***	***97.57***

Với kết quả đạt được như trên, chúng ta thấy rằng với số lượng phần tử trong hai tập $\mathcal{S}$ và $\mathcal{D}$ khác nhau (từ khoảng 5300 đến 9300 phần tử), thuật toán do chúng tôi đề xuất cho kết quả *prec* và *rec* rất cao và ổn định (hơn 97%). Việc thực nghiệm trên bộ dữ liệu chuẩn của PAN được rất nhiều nhóm nghiên cứu và các phòng thí nghiệm trên thế giới sử dụng để đánh giá các phương pháp phát hiện sao chép cũng như sử dụng các độ đo dùng để đánh giá trong các cuộc thi của PAN cho thấy kết quả đạt được hoàn toàn tin cậy để đánh giá các thuật toán, hướng tiếp cận mới do chúng tôi đề xuất.



Hình 3. Giao diện kết quả một lần thử nghiệm cho kết quả *prec* 97.77% và *rec* 97.59%

V. KẾT LUẬN

Trong bài báo này, chúng tôi đã đề xuất mô hình hệ thống, quy trình xử lý và trình bày một cách tiếp cận hoàn toàn mới trong phát hiện sự giống nhau của văn bản, đó là dựa trên phương pháp DWT và bộ lọc Haar. Đóng góp lớn trong bài báo là đề xuất thuật toán để chuyển văn bản thành các chuỗi số thực DNA, xây dựng thuật toán phát hiện sự giống nhau giữa các văn bản. Các kết quả thực nghiệm trên bộ dữ liệu chuẩn của PAN chứng minh thuật toán do chúng tôi đề xuất đem lại hiệu quả cao trong phát hiện sự giống nhau của văn bản.

Trong thời gian tới, chúng tôi sẽ tiếp tục nghiên cứu để tối ưu hơn thuật toán đề xuất và thử nghiệm trên nhiều bộ dữ liệu khác. Đặc biệt là cải tiến thuật toán để áp dụng cho văn bản tiếng Việt đem lại hiệu quả cao nhằm ứng dụng giải quyết bài toán phát hiện sao chép văn bản tiếng Việt.

Từ hướng tiếp cận và phương pháp đề xuất này, chúng tôi cũng đã tính đến xử lý dữ liệu lớn với việc mã hoá dữ liệu văn bản sang dạng tín hiệu số cho phép tìm kiếm nhị phân, vì đây là một trong những phương pháp tìm kiếm nhanh nhất khi làm việc với dữ liệu lớn. Hơn nữa, DWT cho độ phức tạp tính toán chỉ là hàm tuyến tính trong mỗi lần lấy mẫu con nên giải pháp đề xuất sẽ càng hiệu quả trong quá trình xử lý dữ liệu lớn. Chúng tôi sẽ tiến hành phân tích và đo đạc thời gian thực thi trong các nghiên cứu tiếp theo.

LỜI CẢM ƠN

Nghiên cứu này được tài trợ bởi Quỹ Phát triển khoa học và công nghệ Đại học Đà Nẵng trong đề tài có mã số B2017-ĐN01- 07.

TÀI LIỆU THAM KHẢO

[1] Androutsopoulos, I. and P. Malakasiotis, “A survey of paraphrasing and textual entailment methods”, Journal of Artificial Intelligence Research, pp. 135-187, 2010.

[2] Bin-Habtoor, A. and M. Zaher, “A Survey on Plagiarism Detection Systems”, International Journal of Computer Theory and Engineering, vol. 4, no. 2, pp. 185-188, 2012.

[3] Meuschke, N. and B. Gipp, “State-of-the-art in detecting academic plagiarism”, International Journal for Educational Integrity, vol. 9, no. 1, pp. 50-71, 2013.

[4] Đệ, Trần Cao, et al, "Phát triển hệ thống phát hiện đạo văn cho trường đại học Việt Nam", Tạp chí Khoa học Trường Đại học Cần Thơ, vol. 35, pp. 31-39, 2014.

[5] Reddy, G. Suresh, T. V. Rajinikanth, and A. Ananda Rao, “Clustering and Classification of Text Documents Using Improved Similarity Measure”, International Journal of Computer Science and Information Security, vol. 14, pp. 39-54, 2016.

[6] Wahlstrom, S., “Evaluation of String Searching Algorithms”, IDT Mini-conference on Interesting Results in Computer Science and Engineering, 2004.

[7] Lin, Yung-Shen, Jung-Yi Jiang, and Shie-Jue Lee, “A similarity measure for text classification and clustering”, IEEE transactions on knowledge and data engineering, vol. 26, no. 7, pp. 1575-1590, 2014.

[8] Gomaa, W. H. and A. A. Fahmy, “A survey of text similarity approaches”, International Journal of Computer Applications, vol. 68, no. 13, pp. 13-18, 2013.

[9] Mountassir, A., I. Berrada, and H. Benberahim, “Representing text documents in training document spaces: a novel model for document representation”, Journal of Theoretical & Applied Information Technology, vol. 56, no. 1, pp. 30-39, 2013.

[10] Juuso T. Olkkonen (Ed), "Discrete Wavelet Transforms-Theory and Application", 2011.

[11] Vidakovic, B., "Statistical modeling by wavelets", John Wiley & Sons, vol. 503, 2009.

[12] Popivanov, I. and R. J. Miller, "Similarity search over time-series data using wavelets", In Proceedings of the 18th International Conference on Data Engineering (ICDE'02), IEEE, 2002.

[13] Potthast, M., et al, "Overview of the 1st International Competition on Plagiarism Detection", In Stein, B., et al (Ed), PAN'09, pp. 1-9, 2009.

[14] http://www.uni-weimar.de/medien/webis/corpora/corpus-pan-labs-09-today/pan-09/pan09-data/pan09-external-plagiarism-detection-training-corpus-2009-03-30.zip

A NOVEL APPROACH BASED ON DISCRETE WAVELET TRANSFORMATION FOR TEXT SIMILARITY DETECTION

Ho Phan Hieu, Nguyen Thi Ngoc Anh, Nguyen Van Hieu, Dang Thien Binh, Vo Trung Hung

***ABSTRACT**: In this paper, we propose a novel text similarity detection algorithm based on Discrete Wavelet Transform (DWT) approach. In particular, the available source materials are converted into a set of the floating-number sequences, namely source DNAs, which are generated by using DWT. To check the similarity for an arbitrary document, we also apply DWT to derive its own DNAs to which the smallest Euclidean distances from the source DNAs are computed. As compared to a threshold level, the values of these distances indicate whether any piece of the checked document is duplicated from another source. The experimental results demonstrate that the proposed algorithm provides an efficient text similarity detection by testing with a real standard dataset of Plagiarism Analysis, Authorship Identification, and Near-Duplicate detection, known as PAN.*