BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**NIÊN LUẬN CƠ SỞ**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**TÓM TẮT VĂN BẢN BẰNG TIẾNG VIỆT DỰA VÀO KỸ THUẬT VOTING**

**Sinh viên: Nguyễn Tuấn Khanh**

**Mã số: B1706481**

**Khóa: K43**

**Cần Thơ, 10/2020**

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

**NIÊN LUẬN CƠ SỞ**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**TÓM TẮT VĂN BẢN BẰNG TIẾNG VIỆT DỰA VÀO KỸ THUẬT VOTING**

**Người hướng dẫn Sinh viên thực hiện**

**TS Lâm Nhựt Khang Nguyễn Tuấn Khanh**

**Mã số: B1706481**

**Khóa: K43**

***Cần Thơ, 10/2020***

# LỜI CẢM ƠN

Lời đầu tiên, tôi xin bày tỏ sự cảm ơn chân thành đối với TS. Lâm Nhựt Khang – giáo viên hướng dẫn trực tiếp của tôi. Cô Khang đã cho tôi những gợi ý và chỉ dẫn quý báu trong quá trình nghiên cứu và hoàn thiện niên luận cơ sở.

Tôi cũng xin được cảm ơn gia đình, những người thân, và bạn bè tôi đã quan tâm, giới thiệu các tài liệu hữu ích trong thời gian học tập và nghiên cứu niên luận cơ sở.

Mặc dù đã cố gắng hoàn thành niên luận nhưng chắc chắn sẽ không tránh khỏi những sai sót, tôi kính mong nhận được sự thông cảm và chỉ bảo của các thầy cô và các bạn.

Tôi xin chân thành cảm ơn!

**MỤC LỤC**

[LỜI CẢM ƠN i](#_Toc59025114)

[TÓM LƯỢT iv](#_Toc59025115)

[ABSTRACT iv](#_Toc59025116)

[PHẦN GIỚI THIỆU 1](#_Toc59025117)

[CHƯƠNG 1. TỔNG QUAN VỀ TÓM TẮT VĂN BẢN VÀ TÓM TẮT VĂN BẢN TIẾNG VIỆT 3](#_Toc59025118)

[1.1 Giới thiệu về tóm tắt văn bản 3](#_Toc59025119)

[1.1.1 Các giai đoạn của các tham số của hệ thống tóm tắt văn bản 3](#_Toc59025120)

[1.1.2 Tóm tắt văn bản theo hướng rút trích ( Extract ) 4](#_Toc59025122)

[1.2 Hiện trạng nghiên cứu tóm tắt văn bản tiếng Việt 4](#_Toc59025123)

[1.3 Hiện trạng kho ngữ liệu huấn luyện và đánh giá cho bài toán tóm tắt văn bản tiếng Việt 4](#_Toc59025124)

[1.4 Phương pháp đánh giá tóm tắt văn bản 4](#_Toc59025125)

[1.4.1 Đánh giá thủ công 4](#_Toc59025126)

[1.4.2 Đánh giá đồng chọn 5](#_Toc59025127)

[1.4.3 Đánh giá thủ công 6](#_Toc59025128)

[CHƯƠNG 2: PHƯƠNG PHÁP VOTING TRONG TÓM TẮT VĂN BẢN TIẾNG VIỆT 7](#_Toc59025129)

[2.1 Mô hình tóm tắt văn bản sử dụng kỹ thuật Voting 7](#_Toc59025130)

[2.1.1 Mô hình phương pháp Voting 7](#_Toc59025131)

[2.2 Bài toán tóm tắt văn bản tiếng Việt dựa vào kỹ thuật Voting 8](#_Toc59025133)

[2.2 Tính trọng số cậu dựa trên các đặc trưng trong văn bản tiếng Việt 9](#_Toc59025136)

[2.2.1 Trọng số TF.ISF 9](#_Toc59025137)

[2.2.2 Xác suất từ 10](#_Toc59025138)

[2.2.3 Thực thể tên 10](#_Toc59025139)

[CHƯƠNG 3: THỬ NGHIỆM VÀ ĐÁNH GIÁ 12](#_Toc59025140)

[3.1 Tập dữ liệu 12](#_Toc59025141)

[3.2 Tiền xử lý văn bản 12](#_Toc59025142)

[3.3 Phương pháp đánh giá kết quả tóm tắt 12](#_Toc59025143)

[3.4 Lựa chọn các phương pháp tóm tắt văn bản đầu vào 12](#_Toc59025144)

[CHƯƠNG 4: KẾT LUẬN 15](#_Toc59025149)

[TÀI LIỆU THAM KHẢO 16](#_Toc59025150)

[PHỤC LỤC 18](#_Toc59025151)

**DANH MỤC BẢNG VÀ HÌNH VẼ**

[Hình 1-1 Các giai đoạn của hệ thống tóm tắt 3](#_Toc58432794)

[Bảng 1. Minh họa phương pháp Voting. 7](#_Toc58432805)

[Hình 2-1. Mô hình tóm tắt đơn văn bản tiếng Việt dựa vào kỹ thuật Voting. 8](#_Toc58432807)

[Bảng 2. Minh hoạ tóm tắt bằng kỹ thuật Voting 9](#_Toc58432808)

[Bảng 3. Bảng thông kê đặc trưng của 3 phương pháp đầu vào. 13](#_Toc58432818)

[Bảng 4: kết quả tóm tắt của 3 phương pháp 13](#_Toc58432819)

[Bảng 5: Kết quả tóm tắt của mô hình sử dụng kỹ thuật Voting 14](#_Toc58432820)

[Bảng 6: Bảng tổng kết kết quả thử nghiệm trên tập dữ liệu. 14](#_Toc58432821)

# TÓM LƯỢT

Phương pháp tóm tắt văn bản tiếng Việt dựa vào kỹ thuật Voting là một phương pháp mới được đề xuất với ý tưởng xem kết quả của mỗi phương pháp tóm tắt văn bản khác nhau là một lá phiếu có thứ tự ưu tiên. Trong đó mỗi lá phiếu là các câu đã được sắp xếp theo trọng số từ cao xuống thấp. Số lá phiếu có thứ tự sắp xếp câu giống nhau được gọi là hệ số phương pháp, hệ số này được tính toán thông qua học kho ngữ liệu tóm tắt mẫu bằng phương pháp học máy sử dụng giải thuật di truyền. Dựa trên kết quả từng lá phiếu và hệ số phương pháp, sử dụng kỹ thuật Voting để lựa chọn các câu có trọng số cao cho bản tóm tắt cuối cùng.

# ABSTRACT

The Vietnamese text summary method based on the Voting technique is a new proposed method with the idea of considering the result of each different text summary method as an ordered ballot. In which each ballot is the sentences are sorted by weight from high to low. The number of ballots with the same sentence order is called the method coefficient, which is calculated through the linguistic data repository that summarizes the sample by machine learning using genetic algorithms. Based on individual ballot results and methodological coefficients, use the Voting technique to select high weight sentences for the final summary.

# PHẦN GIỚI THIỆU

1. **Đặt vấn đề**

Với sự bùng nổ thông tin lưu trữ trên các hệ thống máy tính và trên Internet, một lượng thông tin khổng lồ được lưu trữ trên đó. Để khai thác hiệu quả lượng thông tin khổng lồ này cần phải có các hệ thống xử lý ngôn ngữ tự nhiên đủ mạnh. Tóm tắt văn bản là một trong những bài toán quan trọng đó.

Bài toán tóm tắt văn bản tiếng Việt đóng một vai trò quan trọng trong việc khai thác hiệu quả thông tin trong kho ngữ liệu văn bản tiếng Việt lớn. Nó có ứng dụng rất lớn trong các hệ thống như: tìm kiếm thông minh, đa ngôn ngữ, tổng hợp thông tin... Đối với lĩnh vực an ninh quốc phòng, tóm tắt tin tức có thể giúp cho cán bộ nghiệp vụ thu thập đủ các thông tin cần thiết và kịp thời theo dõi, đánh giá, xử lý nguồn thông tin một cách nhanh chóng

Vì thế, để giải quyết bài toán này ở đây niên luận đưa ra phương pháp tóm tắt văn bản bằng phương pháp Voting. Ý tưởng của phương pháp này là xem kết quả của mỗi phương pháp tóm tắt văn bản khác nhau là một lá phiếu đã được sắp xếp thứ tự ưu tiên các câu trích rút, sử dụng kỹ thuật Voting để lựa chọn các câu ưu tú dựa trên các lá phiếu đã nêu.

1. **Mục tiêu đề tài**

Đề tài nghiên cứu giải quyết hai vấn đề chính: Một là, trình bày phương pháp tóm tắt đơn văn bản tiếng Việt dựa vào kỹ thuật Voting và hạn chế của nó; hai là, cách đưa hệ số phương pháp vào và xác định bộ hệ số phương pháp bằng giải thuật di truyền thông qua quá trình học tập văn bản tóm tắt mẫu. Sau đó, tiến hành thử nghiệm tóm tắt văn bản và đánh giá chúng.

1. **Đối tượng và phạm vi nghiên cứu**

Nhận thấy đây là một hướng nghiên cứu mới trong tiếng Việt, do đó trong niên luận này sẽ phân tích hạn chế, sau đó nghiên cứu, đề xuất cải tiến bằng cách đưa hệ số phương pháp vào để nâng cao độ chính xác.

1. **Phương pháp nghiên cứu**

* Dựa trên các phương pháp tóm tắt văn bản của thế giới và trong nước.
* Dựa trên phân tích các hạn chế của các phương pháp tóm tắt văn bản tiếng Việt.
* Đề xuất phương pháp tóm tắt văn bản tiếng Việt mới dựa trên một số mô hình toán học phù hợp (phóng sinh học, voting…).
* Kiếm chứng phương pháp đề suất bằng thực nghiệm.

1. **Nội dung nghiên cứu**

Nghiên cứu phương pháp tóm tắt văn bản tiếng Việt sử dụng kỹ thuật Voting: Ý tưởng của phương pháp này là xem kết quả của mỗi phương pháp tóm tắt văn bản khác nhau là lá phiếu đã được sắp xếp thứ tự ưu tiên theo trọng số của các câu (số lá phiếu giống nhau được định nghĩa là hệ số phương pháp được xác định thông quá trình học kho ngữ liệu tóm tắt mẫu), sử dụng kỹ thuật Voting để lựa chọn các câu có trọng số voting cao dựa trên các lá phiếu.

1. **Bố cục niên luận cơ sở**

Niên luận cơ sở gồm 4 chương cùng với các phần giới thiệu, kết luận, phụ lục, tài liệu tham khảo.

Chương 1: Tổng quan về tóm tắt văn bản và tóm tắt văn bản tiếng việt

Chương 2: Phương pháp voting trong tóm tắt văn bản tiếng việt

Chương 3: Thử nghiệm và Đánh giá

Chương 4: Kết luận

# CHƯƠNG 1. TỔNG QUAN VỀ TÓM TẮT VĂN BẢN VÀ TÓM TẮT VĂN BẢN TIẾNG VIỆT

1. **Giới thiệu về tóm tắt văn bản**

Tóm tắt văn bản là quá trình rút trích những thông tin quan trọng nhất từ một hoặc nhiều nguồn để tạo ra phiên bản cô đọng, ngắn gọn phục vụ cho một hoặc nhiều người dùng cụ thể, hay một hoặc nhiều nhiệm vụ cụ thể.

Các nghiên cứu về phương pháp tóm tắt văn bản tập trung vào hai hướng chính: tóm tắt trích rút và tóm tắt tóm lược. Tóm tắt văn bản theo hướng trích rút dễ thực hiện và có tốc độ nhanh hơn so với tóm tắt tóm lược. Hướng tiếp cận tóm tắt trích rút chủ yếu là dựa vào các đặc trưng quan trọng của văn bản để tính trọng số câu để trích rút. Trong khi đó, tóm tắt tóm lược là dựa vào các kỹ thuật xử lý ngôn ngữ tự nhiên kết hợp với thông tin về ngôn ngữ để tạo ra các tóm tắt cuối cùng.

1. **Các giai đoạn của các tham số của hệ thống tóm tắt văn bản**

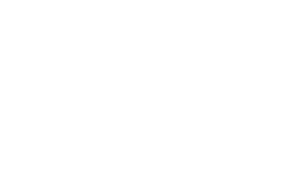
Hệ thống tóm tắt văn bản tự động được chia thành 3 giai đoạn chính:

* + Phân tích (Analysis or Interpretation): Phân tích văn bản đầu vào để đưa ra những mô tả bao gồm các thông tin dùng đế tìm kiếm, đánh giá các đơn vị ngữ liệu quan trọng cũng như các tham số đầu vào cho việc tóm tắt.
  + Biến đổi (Transformation): Lựa chọn các thông tin trích chọn được, biến đổi để giản lược và thống nhất, kết quả là các đơn vị ngữ liệu đã được tóm tắt.
  + Tổng hợp (Synthesis or Realization): Từ các đơn vị ngữ liệu đã tóm tắt, tạo văn bản mới chứa những điểm chính, quan trọng của văn bản gốc.



Văn bản

gốc



Hệ thống tóm tắt văn bản



Phân

tích



Biến

đổi



Tổng

hợp



Văn bản

tóm tắt

# *Hình 1-1 Các giai đoạn của hệ thống tóm tắt*

Các tham số đầu vào: Các đặc trưng của văn bản đầu vào có thể ảnh hưởng tới kết quả tóm tắt theo các yếu tố sau: Cấu trúc văn bản, kích thước, ngôn ngữ, lĩnh vực, đơn vị.

Các tham số mục đích: Các hệ thống tóm tắt tự động có thể tạo ra các bản tóm tắt tổng quát của một văn bản cho trước, hay có thể tạo ra các bản tóm tắt cho một tác vụ được định nghĩa trước. Các yếu tố sau đây có liên quan tới các tham số mục đích của các hệ thống tóm tắt:

* + Tình huống: Tình huống liên quan tới ngữ cảnh của bản tóm tắt
  + Chủ đề: Nếu ta biết trước mối quan tâm của người đọc thì ta có thể tạo ra các bản tóm tắt có liên quan tới chủ đề đó.
  + Mục đích sử dụng: Nếu ta biết trước mối quan tâm của người đọc thì ta có thể tạo ra các bản tóm tắt có liên quan tới chủ đề đó.

1. **Tóm tắt văn bản theo hướng rút trích ( Extract )**

Tóm tắt trích rút (Extract): là một bản tóm tắt báo gồm các đơn vị văn bản quan trọng như câu, đoạn… được trích rút từ văn bản gốc.

1. **Hiện trạng nghiên cứu tóm tắt văn bản tiếng Việt**

Tiếng Việt là ngôn ngữ không biến hình từ và âm tiết tính, tức là mỗi một tiếng (âm tiết) được phát âm tách rời nhau và được thể hiện bằng một chữ viết . Hai đặc trưng này chi phối toàn bộ tổ chức bên trong của hệ thống ngôn ngữ Việt, do vậy trong lĩnh vực xử lý ngôn ngữ tự nhiên tiếng Việt nói chung và bài toán tóm tắt tiếng Việt nói riêng chúng ta cần chú ý tới khi xử lý trên máy tính.

1. **Hiện trạng kho ngữ liệu huấn luyện và đánh giá cho bài toán tóm tắt văn bản tiếng Việt**

Cho đến nay, chưa có một kho ngữ liệu huấn luyện và đánh giá phục vụ cho bài toán tóm tắt văn bản tiếng Việt được công bố. Lý do có thể là do để xây dựng kho ngữ liệu lớn cần một số lượng chuyên gia ngôn ngữ và kinh phí đủ lớn. Việc thiếu kho ngữ liệu huấn luyện và đánh giá cho bài toán tóm tắt văn bản tiếng Việt là một lý do quan trọng để giải thích việc tại sao đến nay các nghiên cứu tóm tắt văn bản tiếng Việt còn ít. Mặt khác, do thiếu kho ngữ liệu huấn luyện và đánh giá mà các phương pháp tóm tắt đã đề xuất cũng chưa được đánh giá so sánh với nhau.

1. **Phương pháp đánh giá tóm tắt văn bản**
2. **Đánh giá thủ công**

Nhà ngôn ngữ học trực tiếp đánh giá bản tóm tắt dựa vào chất lượng bản văn, nghĩa là sử dụng các tham số ngữ pháp, không dư thừa, phân lớp tham chiếu và sự gắn kết để cho điểm bản tóm tắt do hệ thống tạo ra. Cách đánh giá là xem xét lỗi ngữ pháp trong bản văn như sai từ, lỗi dấu câu. Bản tóm tắt khi hệ thống tạo ra không được chứa thông tin dư thừa và các tham chiếu trong bản tóm tắt phải được liên kết rõ ràng với chủ đề của văn bản gốc. Độ gắn kết của văn bản cũng là một tiêu chí quan trọng để đánh giá bản tóm tắt hệ thống. Tuy nhiên, phương pháp này có một số hạn chế như việc chấm điểm do con người thực hiện không ổn định và là phương pháp đánh giá tiêu tốn thời gian và tiền bạc.

1. **Đánh giá đồng chọn**

Phương pháp đánh giá dựa trên đồng chọn chỉ có thể sử dụng với các bản tóm tắt theo hướng trích rút câu. Các câu được trích chọn kết nối với nhau, tạo nên văn bản tóm tắt, không cần hiệu chỉnh thêm. Phương pháp này đánh giá giữa bản tóm tắt do hệ thống trích rút với bản tóm tắt do con người trích rút sử dụng độ đo chính xác (precision), triệu hồi (recall), các giá trị f- measure.

**Độ đo chính xác (precision):** là tỉ số giữa số lượng các câu được cả hệ thống và con người trích rút trên số các câu được hệ thống trích rút.

trong đó: |𝑆𝑀| là số lượng câu của bản tóm tắt do hệ thống trích rút;

|𝑆𝐻| là số lượng câu của bản tóm tắt do con người trích rút;

|𝑆𝐻 ∩𝑆𝑀| là số lượng những câu được cả hệ thống và con người trích rút.

**Độ đo triệu hồi (recall):** là tỉ số giữa số lượng các câu được trích rút bởi hệ thống trùng với số các câu mà con người trích rút trên số các câu chỉ được lựa chọn bởi con người.

trong đó: |𝑆𝑀| là số lượng câu của bản tóm tắt do hệ thống trích rút;

|𝑆𝐻| là số lượng câu của bản tóm tắt do con người trích rút;

|𝑆𝐻 ∩𝑆𝑀| là số lượng những câu được cả hệ thống và con người trích rút.

**Độ đo f-score:** là một độ đo kết hợp hai đại lượng precision và recall. Theo truyền thống thì f-score được định nghĩa là trung bình hàm điều hòa của recision và recall. Các giá trị f-score nhận giá trị trong đoạn [0, 1], trong đó giá trị tốt nhất là 1.

Trong tóm tắt văn bản, người ta cũng thường dùng các trọng số khác nhau cho precision và recall trong khi tính f-score. Giá trị trọng số β là một số thực không âm. Trọng số lớn hơn 1 nghĩa là precision quan trọng hơn, còn trọng số nhỏ hơn 1 nghĩa là recall quan trọng hơn.

Các giá trị precision và recall có thể không phù hợp trong một số trường hợp của tóm tắt văn bản. Ví dụ, từ một văn bản có 5 câu (1, 2, 3, 4, 5), ta tạo ra hai bản tóm tắt khác nhau. Bản tóm tắt thứ nhất chứa các câu (1, 2, 5) và bản kia chứa các câu (1, 4, 5). Bản tóm tắt lý tưởng chứa các câu (1, 2, 5). Khi đánh giá bằng precision và recall, ta có thể quyết định bản tóm tắt đầu tiên tốt hơn bản thứ hai. Nhưng quá trình tóm tắt cũng có tính chủ quan, nên có thể bản tóm tắt thứ hai tốt như bản tóm tắt đầu.

1. **Đánh giá thủ công**

**Phương pháp đánh giá dựa trên LCS:** LCS tìm ra độ dài của chuỗi con chung dài nhất giữa văn bản X và Y, độ dài của chuỗi con chung dài nhất càng lớn thì 2 văn bản X, Y càng giống nhau.

trong đó: 𝑙𝑒𝑛𝑔𝑡ℎ(𝑋) là độ dài của chuỗi X;

𝑙𝑒𝑛𝑔𝑡ℎ(𝑌) là độ dài của chuỗi Y;

là khoảng cách biên tập giữa X và Y (là số lượng tối thiểu của việc xóa và chèn thêm cần thiết để biến đổi X thành Y).

**Phương pháp đánh giá ROUGE:**

Các phương pháp đánh giá tóm tắt truyền thống thường gắn với đánh giá thủ công do chuyên gia con người thực hiện thông qua một số độ đo khác nhau, chẳng hạn: mức độ súc tích, mức độ liền mạch, ngữ pháp, mức độ dễ đọc và nội dung. Tuy nhiên, phương pháp đánh giá kết quả tóm tắt thủ công mất quá nhiều công sức và chi phí. Vì thế, đánh giá tóm tắt tự động là một yêu cầu cấp thiết. Lin và Hovy đề xuất một phương pháp đánh giá mới gọi là ROUGE (Recall-Oriented Understudy for Gisting Evaluation). Hiện nay phương pháp đo này được sử dụng như một phương pháp chuẩn đánh giá kết quả tóm tắt tự động cho văn bản tiếng Anh. Một cách hình thức, ROUGE-N là một độ đo đối với các n-gram trong văn bản tóm tắt hệ thống và trong tập các văn bản tóm tắt lý tưởng, được tính theo công thức:

trong đó: *S* là bản tóm tắt hệ thống; *RSS* là tập văn bản tóm tắt lý tưởng; *Countmatch(n-gram)* là số lượng n-gram đồng xuất hiện lớn nhất giữa văn bản tóm tắt hệ thống và tập văn bản tóm tắt lý tưởng; *Count(n-gram)* là số lượng ngram trong văn bản tóm tắt lý tưởng.

# CHƯƠNG 2: PHƯƠNG PHÁP VOTING TRONG TÓM TẮT VĂN BẢN TIẾNG VIỆT

1. **Mô hình tóm tắt văn bản sử dụng kỹ thuật Voting**
2. **Mô hình phương pháp Voting**

Bầu chọn (voting) là một quá trình đưa ra quyết định lựa chọn một ứng viên trên lá phiếu để chọn ra ứng viên phù hợp cho một mục đích cụ thể. Người ta phân ra thành hai mô hình chính: mô hình chọn một người chiến thắng và mô hình chọn nhiều người chiến thắng. Theo 2 loại mô hình này, có nhiều phương pháp bỏ phiếu khác nhau được đề xuất như: phương pháp số phiếu đồng thuận, phương pháp đa số, phương pháp tính điểm Borda, phương pháp so sánh từng cặp Condorcet, phương pháp Schulze... Mỗi phương pháp có những điểm mạnh yếu riêng và phù hợp với các mô hình chọn ứng viên riêng. Người ta đã xây dựng các tiêu chuẩn riêng cho bài toán bầu cử. Dựa vào các tiêu chuẩn này, tuỳ vào từng mô hình chọn ứng viên (chọn một người chiến thắng hay nhiều người chiến thắng) mà người ta chọn phương pháp bỏ phiếu phù hợp. Chúng ta xem ví dụ trong bảng 1:

*Bảng 1. Minh họa phương pháp Voting.*

|  |  |  |  |
| --- | --- | --- | --- |
|  | Số phiếu bầu chọn | | |
|  | 2 | 3 | 1 |
| Ưu tiên 1 | A | B | C |
| Ưu tiên 2 | B | C | B |
| Ưu tiên 3 | C | A | A |
| Ưu tiên 4 | D | D | D |

Qua phân tích các hướng tiếp cận tóm tắt văn bản theo hướng trích rút hình 1-1. Tôi nhận thấy rằng, các phương pháp theo hướng này đều cho kết quả đầu ra là một danh sách các câu được sắp xếp theo trọng số từ cao đến thấp. Ta có thể xem đây là một lá phiếu bầu cử có thứ tự ưu tiên mà ứng viên là chính là câu. Và bài toán tóm tắt văn bản chính là lựa chọn theo kết quả Voting bằng phương pháp tính điểm Borda.

**Phương pháp tính điểm Borda**

Trong phương pháp này, các vị trí được tính điểm như sau: vị trí cuối cùng được 1 điểm, tiếp theo trên vị trí cuối cùng được 2 điểm,… Sau đó, các điểm được nhân với số phiếu và cộng lại với nhau để có được tổng số điểm cho mỗi ứng viên.

Xét ví dụ ở bảng 1: Ưu tiên 1 được 4 điểm, ưu tiên 2 được 3 điểm, ưu tiên 3 được 2 điểm và ưu tiên 4 được 1 điểm: Điểm cho mỗi ứng cử viên là:

1. = 4×2 + 2×3 + 2×1 = 16
2. = 3×2 + 4×3 + 3×1 = 21
3. = 2×2 + 3×3 + 4×1 = 17
4. = 1×2 + 1×3 + 1×1 = 6

Kết quả bầu chọn từ cao xuống thấp, ta được danh sách theo thứ tự: B, C, A, D.

1. **Bài toán tóm tắt văn bản tiếng Việt dựa vào kỹ thuật Voting**

Ý tưởng của phương pháp này là xem kết quả của mỗi phương pháp tóm tắt văn bản khác nhau là một lá phiếu đã được sắp xếp thứ tự ưu tiên các câu trích rút, sử dụng kỹ thuật Voting để lựa chọn các câu ưu tú dựa trên các lá phiếu đã nêu.

Văn b

ản tiếng Việt đầu

vào

tách câu, tách t

(

ừ, loại bỏ h

ư t

ừ)

K

Ỹ THUẬT

VOTING

L

ấy theo tỉ lệ tóm tắt v

à

xu

ất nguy

ên câu g

ốc

Tóm t

ắt theo

phương pháp (1)

Tóm t

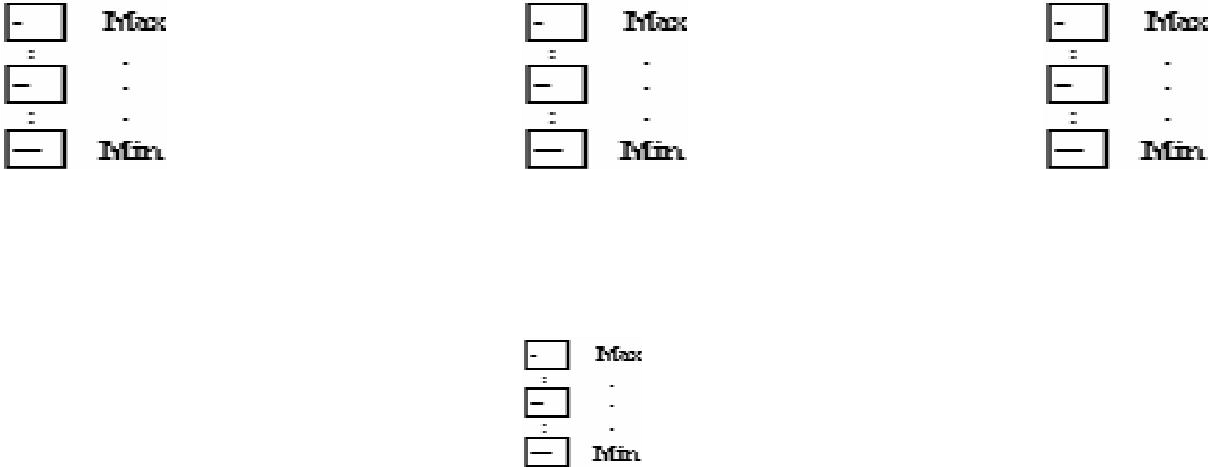
ắt theo

phương pháp (2)

Tóm t

ắt theo

phương pháp (n)



Văn b

ản

tóm t

ắt

*Hình 2-1. Mô hình tóm tắt đơn văn bản tiếng Việt dựa vào kỹ thuật Voting.*

Mô hình tiếp cận sử dụng kỹ thuật Voting cho bài toán tóm tắt đơn văn bản được mô tả như sau:

**Bước 1.** Tiền xử lý văn bản tiếng Việt cho đầu vào: tách câu, tách từ, gán nhãn…

**Bước 2.** Thực hiện tính trọng số câu theo n phương pháp (1), (2),…,(n). Kết quả của mỗi phương pháp là tập các câu được sắp xếp từ cao xuống thấp theo trọng số (Max-Min như trong hình 2-1).

**Bước 3.** Thực hiện kỹ thuật Voting trên tập các kết quả, kết quả Voting là tập các câu được sắp xếp lại theo trọng số Voting từ cao xuống thấp làm kết quả đầu ra.

**Bước 4.** Thực hiện lấy các câu có trọng số Voting từ cao xuống thấp theo tỉ lệ tóm tắt, xuất nguyên các câu đã rút trích theo thứ tự xuất hiện trong văn bản gốc.

**Minh hoạ:** Tóm tắt văn bản có 6 câu với 3 phương pháp tóm tắt làm đầu vào, kết quả của mỗi phương pháp sau khi tính trọng số và sắp xếp từ cao xuống thấp được mô tả dưới bảng 2:

# *Bảng 2. Minh hoạ tóm tắt bằng kỹ thuật Voting*

|  |  |  |  |
| --- | --- | --- | --- |
| **(1)** | **(2)** | **(3)** | **Vongting (Borda)** |
| Câu 2 | Câu 1 | Câu 1 | Câu 1 |
| Câu 1 | Câu 2 | Câu 2 | Câu 2 |
| Câu 6 | Câu 6 | Câu 3 | Câu 6 |
| Câu 3 | Câu 3 | Câu 6 | Câu 3 |
| Câu 5 | Câu 5 | Câu 4 | Câu 5 |
| Câu 4 | Câu 4 | Câu 5 | Câu 4 |

Kết quả chọn theo tỉ lệ 50% sau khi Voting là: Câu 1, Câu 2, Câu 6 (*xem code ở phụ lục trang 18)*

1. **Tính trọng số cậu dựa trên các đặc trưng trong văn bản tiếng Việt**
2. **Trọng số TF.ISF**

**Định nghĩa 2.1:** *Độ quan trọng của câu trong văn bản dựa theo đặc trựng trọng số TF.ISF được tính bằng giá trị trung bình cộng các trọng số TF.ISF của các từ trong câu (được chuẩn hóa về đoạn [0,1]).*

Phương pháp này bắt nguồn từ công thức nổi tiếng *TF.IDF* (Term frequency – inverse document frequency) được sử dụng để xác định mức độ quan trọng của từ trong một văn bản, mà văn bản đó nằm trong một tập hợp các văn bản. Ở đây, niên luận cơ sở tiếp cận bài toán đơn văn bản nên sử dụng trọng số TF.ISF (Term frequency- inverse sentence frequency) làm đặc trưng để tính độ quan trọng của câu.

trong đó: là số các từ có trong câu s;

là từ thứ *k* trong câu s;

là số lần xuất hiện của thực từ trong câu s;

là tần số nghịch của từ trong tập câu thuộc văn bản ( là tổng số câu có trong văn bản; là tổng số câu trong văn bản có chứa thực từ )

Do giá trị của công thức (2.1) tương đối lớn, do vậy giá trị đặc trưng này được chuẩn hóa về khoảng [0,1]. Công thực tính giá trị câu theo đặc trưng TF-ISF được tính theo công thức (2.2):

trong đó : d là văn bản gốc. *(Xem code ở phụ lục trang 18)*

1. **Xác suất từ**

**Định nghĩa 2.2:** *Độ quan trọng của câu trong văn bản dựa theo đặc trưng xác xuất từ được tính bằng giá trị trung bình cộng xác suất của các từ trong câu.*

Đặc trưng này sử dụng xác suất từ để làm nền tảng tính toán trọng số câu. Câu có chứa nhiều từ có tần suất xuất hiện cao trong toàn bộ văn bản thì câu đó càng quan trọng. Công thức tính độ quan trọng của câu tính theo xác suất từ được mô tả như sau:

trong đó: là số các từ có trong câu s;

số lần xuất hiện của từ của câu s trong toàn bộ văn bản;

N là số các từ có trong văn bản. *(Xem code ở phụ lục trang 18)*

1. **Thực thể tên**

Đặc trưng thực thể tên được đưa ra theo quan điểm các thuật ngữ tên riêng (tên người, tên địa danh, tên tổ chức, tên quốc gia…) thường truyền đạt những thông tin quan trọng trong các loại văn bản tin tức. Thực thể tên được xác định thông qua quá trình gán nhãn cho kho ngữ liệu huấn luyện bằng nhãn Np, Ny.

**Định nghĩa 2.3:** Độ quan trọng của câu trong văn bản dựa theo đặc trưng thực thể tên được tính bằng thương của số thực thể tên xuất hiện trong câu và số từ có trong câu.

trong đó: là số thực thể tên xuất hiện trong câu s;

số các từ có trong câu s. *(Xem code ở phụ lục trang 8)*

**CHƯƠNG 3: THỬ NGHIỆM VÀ ĐÁNH GIÁ**

1. **Tập dữ liệu**

Tập dữ liệu sử dụng trong niên luận cơ sở này được xây dựng theo quan điểm thu thập từ những trang báo mạng chính thống. Trong cấu trúc một bài báo mạng thường được chia làm 3 phần: Tiêu đề, tóm tắt, nội dung. Tôi xem phần tóm tắt chính là phần tóm tắt của con người thực hiện. Do vậy, tôi thu thập 1945 văn bản

Độ dài bản tóm tắt hệ thống là 3 câu gần tương đương với bản tóm tắt con người

1. **Tiền xử lý văn bản**

Tập dữ liệu mẫu được lấy từ trang thông tin điện tử nên sẽ còn gặp nhiều vấn đề khó khăn trong quá trình xây dựng và huấn luyện mô hình nên cần phải tiến hành tiền xử lý văn bản.

Ở bước tiền xử lý tôi sử dụng các bộ công cụ sau:

* Sử dụng thực viện Natural Language Toolkit(NLTK) để tiến hang tách câu trong tiếng Việt
* Sử dụng thư viện ViTokenizer để tách từ tiếng Việt
* Sử dụng thư viện ViPosTagger để gán nhãn từ loại

1. **Phương pháp đánh giá kết quả tóm tắt**

Phương pháp đánh giá ROUGE-N đã được đề cập trong mục 1.4.3.

Trong niên luận này tôi sử dụng tool [ROUGE](https://github.com/pltrdy/rouge) để đánh giá.

1. **Lựa chọn các phương pháp tóm tắt văn bản đầu vào**

Trong phần này, để có cơ sở đánh giá hiệu quả của phương pháp Voting. Tôi lựa chọn 03 phương pháp tóm tắt văn bản đầu vào cho phương pháp Voting. Trong đó mỗi phương pháp lựa chọn một số đặc trưng khác nhau để đại diện cho phương pháp, cụ thể: phương pháp 1 chọn 2 đặc trưng trong đó đặc trưng trọng số TF.ISF có vai trò lớn nhất đại diện cho phương pháp 1; phương pháp 2 lại chọn đặc trưng xác suất từ; phương pháp 3 chọn đặc trưng thực thể tên.

Để hiểu rõ hơn, xem bảng mô tả các đặc trưng được lựa chọn của từng phương pháp tóm tắt được thể hiện trong bảng 3:

*Bảng 3. Bảng thông kê đặc trưng của 3 phương pháp đầu vào.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Đặc trưng** | **Phương pháp (1)** | **Phương pháp (2)** | **Phương pháp (3)** |
| F1- Trọng số TF.ISF | x | x |  |
| F2- Xác suất từ |  | x | x |
| F3- Thực thể tên | x |  | x |

**Phương pháp 1:** dựa vào trọng số TF.ISF và thực thể tên.

**Phương pháp 2:** dựa vào trọng số TF.ISF và xác suất từ.

**Phương pháp 3:** dựa vào xác suất từ và thực thể tên.

Với 3 phương pháp nêu trên, tiến hành thực hiện tóm tắt văn bản. Kết quả của 3 phương pháp được mô tả trong bảng 4:

# *Bảng 4: kết quả tóm tắt của 3 phương pháp*

|  |  |  |
| --- | --- | --- |
| **Phương pháp** | **Kết quả tóm tắt (ROUGE-N)** | |
| N=1 | N=2 |
| Phương pháp (1) | 0.264 | 0.123 |
| Phương pháp (2) | 0.263 | 0.120 |
| Phương pháp (3) | 0.268 | 0.127 |

Tiếp theo, tôi tiến hình Voting trên 3 phương pháp là phương pháp (1)(2)(3), như ta quan sát trong bảng 4, mỗi phương pháp đều sử dụng đặc trưng có vái trò cao riêng làm chủ đạo cho phương pháp. Cụ thể: phương pháp(1) sử dụng đặc trưng trọng số TF.ISF, phương pháp (2) sử dụng đặc trưng xác suất từ, phương pháp (3) sử dụng đặc trưng thực thể tên.

Kết quả của mô hình tóm tắt văn bản sử dụng kỹ thuật Voting được mô tả dưới bảng 5:

# *Bảng 5: Kết quả tóm tắt của mô hình sử dụng kỹ thuật Voting*

|  |  |  |
| --- | --- | --- |
| **Phương pháp** | **Kết quả tóm tắt (ROUGE-N)** | |
| N=1 | N=2 |
| **Voting 3 phương pháp (1)(2)(3)** | 0.270 | 0.127 |

Với các kết quả, ta có bảng 6 tổng hợp kết quả tóm tắt:

# *Bảng 6: Bảng tổng kết kết quả thử nghiệm trên tập dữ liệu.*

|  |  |  |
| --- | --- | --- |
| **Phương pháp** | **Kết quả tóm tắt (ROUGE-N)** | |
| N=1 | N=2 |
| Phương pháp (1) | 0.264 | 0.123 |
| Phương pháp (2) | 0.263 | 0.120 |
| Phương pháp (3) | 0.268 | 0.127 |
| **Voting 3 phương pháp (1)(2)(3)** | 0.270 | 0.127 |

Quảng bảng tổng kết trên, cho thấy:

Trên tập dữ liệu thử nghiệm với kết quả của 3 phương pháp khá cạnh tranh nhau thì kết quả Voting cao hơn các phương pháp đơn lẻ. Vậy có thể kết luận rằng, phương pháp tóm tắt văn bản tiếng Việt theo hướng trích rút sử dụng kỹ thuật Voting cho kết quả tốt hơn các phương pháp tóm tắt đơn lẻ với tỉ lệ là 0.27.

# CHƯƠNG 4: KẾT LUẬN

Bài toán tóm tắt tiếng Việt có ý nghĩa quan trọng trong nâng cao hiệu quả khai thác thông tin từ các kho ngữ liệu văn bản tiếng Việt. Các công cụ tóm tắt tiếng Việt được ứng dụng nhiều trong các hệ thống tìm kiếm thông minh, đa ngôn ngữ, tổng hợp thông tin... Đối với lĩnh vực an ninh quốc phòng, tóm tắt tin tức có thể giúp cho cán bộ nghiệp vụ thu thập đủ các thông tin cần thiết và kịp thời theo dõi, đánh giá, xử lý nguồn thông tin một cách nhanh chóng. Nâng cao hiệu quả và độ chính xác của tóm tắt tiếng Việt là hướng nghiên cứu có ý nghĩa khoa học và thực tiễn luôn được các nhà khoa học quan tâm nghiên cứu. Chính vì vậy, mục tiêu nghiên cứu của luận án này là đề xuất các phương pháp tóm tắt văn bản mới phù hợp với văn bản tiếng Việt, có thể áp dụng xây dựng các phần mềm tóm tắt văn bản tiếng Việt chất lượng cao phục vụ trong nhiều lĩnh vực, nhất là lĩnh vực an ninh quốc phòng.

Trong niên luận cơ sở này tôi đã khái quát trình bày các bước cơ bản để xây dựng bài toán tóm tắt văn bản trong nâng cao hiệu quả khai thác thông tin từ các kho ngữ liệu văn bản tiếng Việt. Xây dựng bài toán tóm tắt văn bản theo hướng rút trích ES(Extraction Summarization) là thông qua tính toán các đặc trưng tần suất từ, thực thể tên, trọng số TF.ISF,… để chọn ra các câu quan trọng nhất theo tỉ lệ trích rút.

Phương pháp tóm tắt văn bản mới dựa theo kỹ thuật Voting với ý tưởng xem kết quả của mỗi phương pháp tóm tắt văn bản khác nhau là một lá phiếu đã được sắp xếp thứ tự ưu tiên các câu.Kết quả tóm tắt văn bản theo kỹ thuật Voting đạt kết quả khá khiếm tốn với 27%. Tuy nhiên, với phương pháp Voting còn gặp nhiều hạn chế. Những mặc hạn chế đó là nếu ta xem mỗi phương pháp là một lá phiếu thì có khả năng xảy ra là số phương pháp yếu nhiều hơn sẽ thắng số phương pháp tốt (theo quan điểm đa số) và ngược lại. Điều đó làm cho kết quả Voting có thể không tốt,… Ngoài ra, đề tài còn hạn chế về tập dữ liệu thu thập từ các trang báo điện tử và việc lấy phần tóm tắt con người để đánh giá còn nhiều khó khăn. Chính vì vậy, thách thức đặt ra để mở rộng đề tài này theo hướng tiếp cận tiếp theo là làm sao để giải quyết vấn đề các phương pháp yếu sẽ thắng các phương pháp tốt,cũng như cần mở rộng tập dữ liệu. Có thể kết hợp phương pháp Voting với giải thuật di truyền hay giải thuật tối ưu đàn kiến.

# TÀI LIỆU THAM KHẢO

**Tiếng Việt**

|  |  |
| --- | --- |
| [1] | D. Q. Ban, Ngữ Pháp Tiếng Việt, NXB giáo dục, 2006. |
| [2] | T. N. A. Nguyễn Nhật An, “Tóm tắt văn bản tiếng Việt dựa vào kỹ thuật Voting” trong *Chuyên san Công nghệ thông tin và Truyền thông (JICT) thuộc Tạp chí Khoa học và Kỹ thuật*, Học viện Kỹ thuật quân sự, tháng 4/2014, 2014, pp. pp.57-67. |
| [3] | N. V. Tu, Từ điển đồng nghĩa Tiếng Việt,, NXB giáo dục., 2001. |
| [4] | H. K. Đỗ Phúc, Rút trích ý chính từ văn bản tiếng Việt, Tạp chí Công nghệ Thông tin và Truyền thông, 2006. |
| [5] | L. T. Hương, Nghiên cứu một số phương pháp tóm tắt văn bản, Báo cáo tổng kết đề tài cấp KH và CN cấp bộ, Đại học Bách khoa Hà Nội, 2014. |
| [6] | N. T. T. Hà, “Phát triển một số thuật toán tóm tắt văn bản Tiếng Việt sử dụng phương pháp học bán giám sát,” trong *Luận án Tiến sỹ*, Học viện, 2012. |
| [7] | H. Phê, Từ điển tiếng Việt, NXB giáo dục, 1998. |
| [8] | N. H. T. A. Luận án Tiến sỹ, Tiếp cận đồ thị biểu diễn, khai thác văn bản và ứng dụng, Trường Đại học Khoa Học Tự Nhiên, ĐHQG-HCM, 2011. |
| [9] | T. N. A. P. T. N. H. Nguyễn Nhật An, “Kỹ thuật Voting trong bài toán tách từ tiếng Việt,” trong *Tạp chí NCKH&CNQS*, Đặc san CNTT 04/2014, pp. tr.54-61. |
| [10] | N. Q. D. Trương Quốc Định, “Một giải pháp tóm tắt văn bản tiếng Việt tự động,” trong *Hội thảo quốc gia lần thứ XV: Một số vấn đề chọn lọc của Công nghệ*, 03-04/12/2012. |
| [11] | N. T. T. H. Luận án Tiến sỹ, “Phát triển một số thuật toán tóm tắt văn bản Tiếng Việt sử dụng phương pháp học bán giám sát,” trong *HV KTQS*, 2012, p. 175 trang. |
| [12] | T. M. Vũ, "Tóm tắt đa văn bản dựa vào trích xuất câu," in *Luận văn Thạc sỹ*, Trường ĐHCN, Đại học Quốc gia Hà Nội, 2010.  **Tiếng Anh** |
| [13] | S. A. X. H. P. T. B. H. H. S. M.L. Nguyen, "Sentence Extraction with Support Vector Machine Ensemble," in *Proceedings of the First World Congress of the International Federation for Systems Research : The New Roles of Systems Sciences For a Knowledge-based Society*, 2005. |
| [14] | P. T. A. T. C. D. N. X. H. Nguyen Quang Uy, "A Study on the Use of Genetic Programming for Automatic Text Summarization," in *Conference on Knowledge and Systems Engineering*, KSE, 2012 4th Inter, 2012, pp. pp.93-98. |
| [15] | V. project, "Vietnamese Language Processing," [Online]. Available: http://vlsp.vietlp.org. |
| [16] | E. Pacuit, "Voting Methods," The Stanford Encyclopedia of Philosophy , 2012. [Online]. Available: https://plato.stanford.edu/entries/voting-methods/. |
| [17] | R. a. B. R.-N. Baeza-Yates, Modern, Addison Wesley, 1999. |
| [18] | D. R. e. a. Radev, "Evaluation Challenges in Large-scale Document Summarization," in *ACL '03 Proceedings of the 41st Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics*, 2003, pp. pp.375-382. |
| [19] | S. G. a. B. C., "Term-weighting approaches in automatic text retrieval," in *Information Processing and Management 24, 1988.513-523.*, Reprinted in: Sparck-Jones, K.; Willet, P. (eds.) Readings in, pp. 323-328. |

# PHỤC LỤC

**GIAI ĐOẠN TIỀN XỬ LÝ VĂN BẢN**

import nltk

from pyvi import ViTokenizer, ViPosTagger

import numpy as np

import os

from string import punctuation

from tqdm import tqdm

class load\_data:

def read\_data\_train(path):

# xử lý văn bản

with open(path, 'r', encoding='utf-8') as f:

data = f.read()

find\_content = data.find("Content:")

doc = data [find\_content + 8: ]

sentences = nltk.sent\_tokenize(doc)

X = []

for sentence in sentences:

sent=[]

sentence\_tokenized = ViTokenizer.tokenize(sentence)

for word in sentence\_tokenized.split(" "):

if (word not in list(punctuation)):

sent.append(word)

X.append(" ".join(sent))

return X

def read\_data\_test(path):

# xử lý văn bản

f = open(path, 'r', encoding='utf-8')

data = f.read()

find\_summary = data.find("Summary:")

find\_content = data.find("Content:")

doc = data[find\_summary + 8: find\_content]

sentences = nltk.sent\_tokenize(doc)

X = []

for sentence in sentences:

sent=[]

sentence\_tokenized = ViTokenizer.tokenize(sentence)

for word in sentence\_tokenized.split(" "):

if (word not in list(punctuation)):

sent.append(word)

X.append(" ".join(sent))

return X

def tao\_danh\_sach(document):

# list các từ có trong văn bản

list\_word = []

# mang lưu số từ của 1 câu

word\_in\_sentence = []

# danh sách các từ trong câu

list\_sentence = []

for cau in document:

#cau=cau.lower()

cau = cau.split(" ")

word\_in\_sentence.append(len(cau))

list\_sentence.append(cau)

for word in cau:

list\_word.append(word)

return list\_word, word\_in\_sentence, list\_sentence

**TÍNH CÁC ĐẶC TRƯNG**

import numpy as np

**Trọng số TF.ISF**

def tinh\_TF\_ISF(sentence):

# tạo danh sách số từ trong văn bản,số từ của 1 câu,số câu trong văn bản

list\_word, word\_in\_sentence, list\_sentence = load\_data.tao\_danh\_sach(sentence)

# Tính TF

TF = np.array(word\_in\_sentence)

# -------------Tính ISF------------------------

SF\_dict = dict.fromkeys(list\_word, 0)

# tổng số câu (SF) có chứa từ X

for key in SF\_dict.keys():

count = 0

for n\_cau in list\_sentence:

dem\_in\_cau = n\_cau.count(key)

if(dem\_in\_cau > 0):

count += 1

SF\_dict[key] = count

# tinh ISF

import math

ISF\_word\_dict = {}

for k, v in SF\_dict.items():

s\_f = math.log10(len(list\_sentence)/v)

ISF\_word\_dict[k] = s\_f

#tinh ISF của câu

ISF\_sentence = []

for n\_cau in list\_sentence:

tong = 0

for k, v in ISF\_word\_dict.items():

if(k in n\_cau):

if(n\_cau.count(k) > 1):

v = v\*2

tong += v

ISF\_sentence.append(tong)

# tính trọng số

trong\_so\_TFISF=(np.array(TF)\*np.array(ISF\_sentence))/np.array(word\_in\_sentence)

# trọng số của 1 câu tính theo TF-ISF

if (max(trong\_so\_TFISF)):

trong\_so\_cau = np.array(trong\_so\_TFISF)/max(trong\_so\_TFISF)

else: trong\_so\_cau = np.array(trong\_so\_TFISF)

return trong\_so\_cau

**Xác suất từ**

#build từ điển số lần suất hiện của từ trong văn bản

def word\_dict(sentences,list\_word):

#tạo từ điển từ

wordict=dict.fromkeys(list\_word,0)

#số lần xuất hiện của từ trong toàn bộ văn bản

for n\_cau in sentences:

#n\_cau=n\_cau.lower()

for tu in n\_cau.split(" "):

wordict[tu]+=1

return wordict

#tống số lần xuất hiện

def sum\_lan\_xuat\_hien\_tu\_in\_cau(wordict,list\_sentence):

solanxuathien=[]

for sentence in list\_sentence:

dem=0

for words, count in wordict.items():

if (words in sentence):

if(sentence.count(words) > 1):

count=count\*2

dem+=count

solanxuathien.append(dem)

return solanxuathien

def xac\_suat\_tu(sentences):

# tạo danh sách số từ trong văn bản,số từ của 1 câu,số câu trong văn bản

list\_word, word\_in\_sentence, list\_sentence = load\_data.tao\_danh\_sach(sentences)

#tạo từ điển

wordict=xac\_suat.word\_dict(sentences,list\_word)

#tính tổng lần xuất hiện của 1 từ

solanxuathien=xac\_suat.sum\_lan\_xuat\_hien\_tu\_in\_cau(wordict,list\_sentence)

#tính xác xuất thực thực từ

trongso=np.array(solanxuathien)/len(list\_word)/np.array(word\_in\_sentence)

return trongso

**Thực thể tên**

def tinh\_thuc\_the\_ten(sentences):

X = []

for sent in sentences:

sent = ViPosTagger.postagging(sent)

X.append(sent)

list\_cau = []

list\_nhan = []

for x in X:

for y in range(len(x)):

if(y % 2 == 0):

list\_cau.append(x[y])

else:

list\_nhan.append(x[y])

# tính số thực thể tên có trong câu s

N\_name = []

Nw = [] # số thực từ có trong câu

for loai in list\_nhan:

lent = len(loai)

if("Np" in loai) or ("Ny" in loai):

dem\_in= loai.count('Np')+loai.count('Ny')

else: dem\_in = 0

N\_name.append(dem\_in)

Nw.append(lent)

# tính trọng số câu dựa vào thực thể tên

trong\_so\_cau = np.array(N\_name)/np.array(Nw)

return trong\_so\_cau

**MODEL VOTING**

import numpy as np

from tienxulyvannban import load\_data

from thuc\_the\_ten import danh\_tu\_rieng

from tf\_isf import TF\_ISF

from xac\_suat\_tu import xac\_suat

import os

#---------TÍNH CÁC PHƯƠNG PHÁP ------------------------#

def phuong\_phap(dac\_trung\_1, dac\_trung\_2,data):

PP = np.array(dac\_trung\_1)\*np.array(dac\_trung\_2)

# sắp xếp

sentence\_dict = dict.fromkeys(data, 0)

n = 0

for cau in sentence\_dict.keys():

if(n < len(PP)):

sentence\_dict[cau] = PP[n]

n += 1

PP\_dict = {}

for ts in sorted(PP, reverse=True):

for cau\_n, gt in sentence\_dict.items():

if (gt == ts):

PP\_dict[cau\_n] = gt

sort\_PP = []

for key in PP\_dict.keys():

sort\_PP.append(key)

## tóm tắt văn bản bằng phương pháp :

cau\_pp = []

so\_cau\_tom\_tat = 3

for key in PP\_dict.keys():

if(len(cau\_pp) < so\_cau\_tom\_tat):

cau\_pp.append(key)

tom\_tat = []

for cau\_n in data:

if(cau\_n in cau\_pp):

cau\_n=cau\_n.replace("\_"," ")

tom\_tat.append(cau\_n)

return sort\_PP, tom\_tat

#------------------------------TÍNH THỨ TỰ ƯU TIÊN---------------------

def uu\_tien(phuong\_phap\_n,sentence):

thu\_tu = []

n = len(phuong\_phap\_n)

for sent in sentence:

if sent in phuong\_phap\_n:

thu\_tu.append(n-phuong\_phap\_n.index(sent))

return thu\_tu

# -------------------MODEL VOTING(BORDA)-------------------------------------------

def model\_voting(pp1,pp2,pp3,data\_train):

thu\_tu\_1 = Voting.uu\_tien(pp1,data\_train)

thu\_tu\_2 = Voting.uu\_tien(pp2,data\_train)

thu\_tu\_3 = Voting.uu\_tien(pp3,data\_train)

voting = np.array(thu\_tu\_1) + np.array(thu\_tu\_2) + np.array(thu\_tu\_3)

voting\_dict = dict.fromkeys(data\_train, 0)

n = 0

for cau in data\_train:

if n < len(voting):

voting\_dict[cau] = voting[n]

n += 1

cau\_dict = {}

for ts in sorted(voting, reverse=True):

for cau, gt in voting\_dict.items():

if (gt == ts):

cau\_dict[cau] = gt

sort\_cau\_voting=[]

so\_cau\_tom\_tat = 3 #số câu cần tóm tắt

for key in cau\_dict.keys():

if(len(sort\_cau\_voting) < so\_cau\_tom\_tat):

sort\_cau\_voting.append(key)

tom\_tat = []

for cau\_n in data\_train:

if(cau\_n in sort\_cau\_voting):

cau\_n=cau\_n.replace("\_"," ")

tom\_tat.append(cau\_n)

return tom\_tat

#hàm chạy model

def run\_model( train\_paths,dir\_path):

count = 0

for f in os.listdir(train\_paths):

path\_dir =os.path.join(train\_paths,f)

if os.path.exists(dir\_path +"\\voting summary") ==0 :

os.mkdir(dir\_path +"\\voting summary")

os.mkdir(dir\_path +"\\voting summary\\"+ f)

if os.path.exists(dir\_path +"\\pp1 summary") ==0:

os.mkdir(dir\_path +"\\pp1 summary")

os.mkdir(dir\_path +"\\pp1 summary\\"+f)

if os.path.exists(dir\_path +"\\pp2 summary") ==0:

os.mkdir(dir\_path +"\\pp2 summary")

os.mkdir(dir\_path +"\\pp2 summary\\"+f)

if os.path.exists(dir\_path +"\\pp3 summary") ==0:

os.mkdir(dir\_path +"\\pp3 summary")

os.mkdir(dir\_path +"\\pp3 summary\\"+f)

for path in os.listdir(path\_dir):

if path != ".DS\_Store":

file\_path = os.path.join(path\_dir, path)

for data in os.listdir(file\_path):

sentences = []

txt = os.path.join(file\_path, data)

data = load\_data.read\_data\_train(txt)

for sent in data:

sentences.append(sent)

tf\_isf = TF\_ISF.tinh\_TF\_ISF(sentences)

xac\_suat\_tu = xac\_suat.xac\_suat\_tu(sentences)

thuc\_the\_ten = danh\_tu\_rieng.tinh\_thuc\_the\_ten(sentences)

sort\_pp1,tom\_tat\_pp1 = Voting.phuong\_phap(tf\_isf, thuc\_the\_ten,sentences)

sort\_pp2,tom\_tat\_pp2 = Voting.phuong\_phap(tf\_isf, xac\_suat\_tu,sentences)

sort\_pp3,tom\_tat\_pp3 = Voting.phuong\_phap(thuc\_the\_ten, xac\_suat\_tu,sentences)

tom\_tat\_voting = Voting.model\_voting(sort\_pp1,sort\_pp2,sort\_pp3,sentences)

count +=1

with open("./data/voting summary/"+ f + "/"+str(count) + ".txt",'w', encoding='utf-8') as f1:

f1.write(".\n".join(tom\_tat\_voting))

with open("./data/pp1 summary/"+ f + "/"+str(count) + ".txt",'w', encoding='utf-8') as f2:

f2.write(".\n".join(tom\_tat\_pp1))

with open("./data/pp2 summary/"+ f + "/"+str(count) + ".txt",'w', encoding='utf-8') as f3:

f3.write(".\n".join(tom\_tat\_pp2))

with open("./data/pp3 summary/"+ f + "/"+str(count) + ".txt",'w', encoding='utf-8') as f4:

f4.write(".\n".join(tom\_tat\_pp3))

**ĐÁNH GIÁ ROUGE**

import os

from rouge import Rouge

from model\_Voting import Voting

**#** hàm đánh giá

def danh\_gia(summary\_path,test\_paths):

tamp1 = dict.fromkeys(['f','p','r'],0)

tamp2 = dict.fromkeys(['f','p','r'],0)

tamp3 = dict.fromkeys(['f','p','r'],0)

avg\_scores ={}

count = 0

for paths in os.listdir(summary\_path):

path = os.path.join(summary\_path,paths)

for file\_summary in os.listdir(path):

file\_summarys = os.path.join(path,file\_summary)

summary = open(file\_summarys , encoding='utf-8').read()

for file\_tests in os.listdir(test\_paths):

if file\_tests == paths:

file\_test = os.path.join(test\_paths,file\_tests)

for test\_summary in os.listdir(file\_test):

if test\_summary == file\_summary:

test\_summary = os.path.join(file\_test,test\_summary)

test = open(test\_summary , encoding='utf-8').read()

scores = rouge.get\_scores(summary,test)

for key in scores:

for indx , val in key.items():

if indx =="rouge-1":

for var,gt in val.items():

tamp1[var] +=gt

avg\_scores[indx]=tamp1

if indx=="rouge-2":

for var,gt in val.items():

tamp2[var] +=gt

avg\_scores[indx] =tamp2

if indx=="rouge-l":

for var,gt in val.items():

tamp3[var] +=gt

avg\_scores[indx] =tamp3

count +=1

for index ,val in avg\_scores.items():

for ind1,val1 in val.items():

avg\_scores[index][ind1] = val1 / float(count)

return avg\_scores

#------ hàm main ----------------------

dir\_path = os.path.dirname(os.path.realpath(os.getcwd()))

dir\_path = os.path.join(dir\_path, 'module voting\\data')

test\_paths = os.path.join(dir\_path,'test')

train\_paths = os.path.join(dir\_path, 'train')

summary\_voing\_path = os.path.join(dir\_path, 'voting summary')

summary\_pp1\_path = os.path.join(dir\_path, 'pp1 summary')

summary\_pp2\_path = os.path.join(dir\_path, 'pp2 summary')

summary\_pp3\_path = os.path.join(dir\_path, 'pp3 summary')

# chạy model tạo folder summary của từng phương pháp

Voting.run\_model(train\_paths,dir\_path)

#---------------------thực hiện đánh giá ROUGE ------------------------

rouge = Rouge()

# điểm đánh giá của từng phương pháp

pp1 = danh\_gia(summary\_pp1\_path,test\_paths)

pp2 = danh\_gia(summary\_pp2\_path,test\_paths)

pp3 = danh\_gia(summary\_pp3\_path,test\_paths)

voting = danh\_gia(summary\_voing\_path,test\_paths)

print(pp1)

print("\n")

print(pp2)

print("\n")

print(pp3)

print("\n")

print(voting)