

**ĐẠI HỌC QUỐC GIA TP.HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

A blue logo with a flower and a planet

Description automatically generated

**BÁO CÁO ĐỒ ÁN**

**ĐỀ TÀI: PHÂN LOẠI CHỦ ĐỀ CỦA BÀI BÁO**

|  |  |  |
| --- | --- | --- |
| **Giảng viên hướng dẫn** | **:** | **Nguyễn Thanh Sơn** |
| **Môn học** | **:** | **Kỹ thuật lập trình python** |
| **Sinh viên thực hiện** | **:** | **Nguyễn Quốc Thái** |
|  |  |  |

Thành phố Hồ Chí Minh, ngày 30 tháng 4 năm 2024

|  |  |
| --- | --- |
| **Trường Đại học công nghệ thông tin**  **Đại học quốc gia TP. Hồ Chí Minh**  **Khoa khoa học kỹ thuật và thông tin**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập – Tự do – Hạnh phúc**  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** |

**Thực hiện đề tài:** Phân loại chủ đề của bài báo.

**STT Họ tên SV MSSV**

1 Nguyễn Quốc Thái 19520936

**NHẬN XÉT CỦA GIẢNG VIÊN**

*Tp. Hồ Chí Minh, ngày 30, tháng 4, năm 2024*

**Giảng viên hướng dẫn**

**Nguyễn Thanh Sơn**

1. **GIỚI THIỆU ĐỒ ÁN**

**1 - Mục đích nghiên cứu**

Trong quá trình viết bài của các nhà báo, blogger, nhà sáng tạo nội dung hay đơn giản chỉ là các quản trị viên upload các bài viết lên một trang web hoặc một mạng xã hội nào đó thì việc gắn nhãn cho bài viết đó như thế nào để hướng tới người đọc trong khi có nhiều nội dung được đan xen với nhau trong bài viết được tạo ra.

Chính vì thế mà mục đích của đề tài này ra đời nhằm giải quyết việc phân loại dán nhãn cho bài viết trước khi chúng được xuất bản đăng tải.

**2 - Đối tượng hướng tới**

Tất cả các loại văn bản được viết sử dụng ngôn ngữ tiếng Việt của chúng ta hiện nay: kinh tế, xã hội, thể thao, ..... Khi đó ứng với mỗi bài viết chúng ta cần gắn nhãn cho chúng với chủ đề nào đó mà đôi khi ta tưởng chừng như đơn giản mà lại khó khăn.

**3 - Ứng dụng của đồ án**

Gắn thẻ nội dung hoặc sản phẩm: bằng cách sử dụng danh mục như cách để cải thiện trình duyệt hoặc để xác định nội dung liên quan trên các trang web của bạn. Các nền tảng như thương mại điện tử, cơ quan tin tức, người quản lý nội dung, blog hay thư mục có thể sử dụng công nghệ tự động này để phân loại và gắn thẻ nội dung sản phẩm.

Sử dụng trong tự động hóa các tác vụ CRM (Customer relationship management). Trình phân loại văn bản có khả năng tùy biến cao và có thể được đào tạo phù hợp. Các nhiệm vụ CRM có thể được chỉ định và phân tích trực tiếp dựa trên tầm quan trọng và mức độ phù hợp. Giúp giảm thiểu công việc thủ công hiệu quả về thời gian.

Tối ưu hóa SEO. Tự động hóa gắn thẻ nội dung trên các trang web và giúp người dùng tốt hơn, giúp các nhà tiếp thị nghiên cứu phân tích thẻ nhãn dán giúp tiếp cận người dùng hay nâng cao khả năng cạnh tranh cho các sản phẩm của họ.

1. **QUÁ TRÌNH THỰC HIỆN**

**1 – Tuần 1: Thu thập dữ liệu cho bài toán**

Dữ liệu là yếu tố quan trọng nhất và cũng là các vấn đề mà chúng ta cần quan tâm tới. Trong bất kỳ xây dựng một mô hình dữ liệu nào bước chuẩn bị dữ liệu đề quan trọng dẫn tới thành bại của quá trình xây dựng.

Với bài toán phân loại dán nhãn văn bản tiếng Việt, dữ liệu mà chúng ta cần chuẩn bị là các dữ liệu bài báo tiếng việt kèm theo các chủ đề của nó.

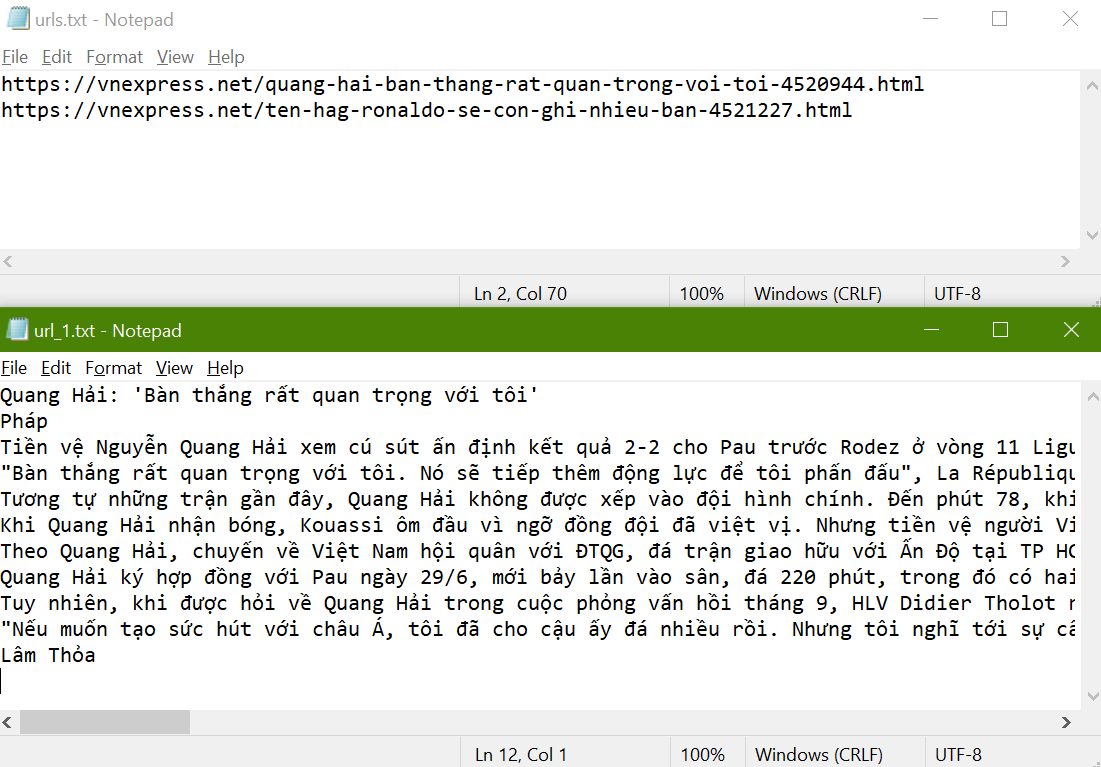
Trong tuần 1, em đã tiến hành cào dữ liệu từ các VNExpress, Vietnamnet, Dantri để phục vụ cho bài toán phân loại.

A screenshot of a computer

Description automatically generated

Hình 1 - Các đoạn văn bản cần cào

Tại đây em sử dụng phương pháp cào url cho trước. Ta nhập Url vào một file txt rồi gọi đường dẫn file chương trình để tiến hành cào.



Hình 2 – Ví dụ cho file txt chứa url và nội dung file

Ngoài ra, em cũng hỗ trợ cào bài báo theo thể chủ đề. Ở đây em hỗ trợ cào bài báo theo 18 chủ để như sau:

* Công nghệ
* Du lịch
* Giáo dục
* Giải trí
* Kinh doanh
* Nhịp sống
* Phim ảnh
* Pháp luật
* Sống trẻ
* Sức khỏe
* Thế giới
* Thể thao
* Thời sự
* Thời trang
* Xe 360
* Xuất bản
* Âm nhạc
* Ẩm thực

Link dữ liệu đã cào : https://drive.google.com/file/d/168SAc7d2A\_ZF6iiaZoJ2ediCsbECcZjR/view?usp=sharing

**2 – Tuần 2: Xử lý dữ liệu dành cho Train / Test model**

Trong tuần 2, em đã tiến hành xử lý dữ liệu dành cho train / test model của mình. Cụ thể hơn em sẽ giải thích bằng các hình ảnh được chụp từ đồ án.

Bước 1: Giải nén Google Drive trên Colab.

A screen shot of a computer

Description automatically generated

Hình 3 – Giải nén google drive trên colab.

Bước 2: Tìm kiếm link của dữ liệu trên Colab.

A screenshot of a computer

Description automatically generated

Hình 4 - Lấy link dữ liệu cần train từ Google Colab.

Bước 3: Giải nén dữ liệu

A screen shot of a computer

Description automatically generated

Hình 5 - Giải nén dữ liệu từ link Google Colab.

Bước 4: Xác định xem giải nén thành công và dữ liệu tồn tại trong trường trình

A screenshot of a computer screen

Description automatically generated

Hình 6 - Xác định tồn tại dữ liệu.

Bước 5: Thống kê số lượng dữ liệu data theo

A screenshot of a computer program

Description automatically generated

Hình 7 - Thống kê data theo nhãn.

Bước 6: Thống kê các từ (word) tồn tại tất cả các nhãn.

A computer screen shot of a program

Description automatically generated

Hình 8 - Thống kê các từ tổn tại trong tất cả các nhãn

Bước 7: Loại bỏ các từ stop-word và lưu file.

Stopword: là các từ ngữ bị một số công cụ tìm kiếm bỏ qua một phần hoặc hoàn toàn. VD: "a, of, many, ..." chiếm khoảng 25% trong các bài viết. Bởi vì các từ ngữ này không liên quan gì tới nội dung và ý nghĩa của các bài viết.

A computer screen shot of a program

Description automatically generated

Hình 9 - Loại bỏ các StopWord

Bước 8: Chia các tập để Train/Test dữ liệu

A computer screen shot of a program code

Description automatically generated

Hình 10 - Chia tập train/test

Bước 9: Lưu train/test thành các file train/test

A computer screen shot of a program code

Description automatically generated

Hình 11 - Lưu các file Train/Test

Bước 10: Khởi tạo đường dẫn Model vào chương trình

A screen shot of a computer

Description automatically generated

Hình 12 - Khởi tạo đường dẫn Model

Bước 11: Khai thác thuật toán Naive Bayes

Link hỗ trợ: [sklearn.naive\_bayes.MultinomialNB — scikit-learn 1.0.1 documentation](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html)

A computer screen shot of a program

Description automatically generated

Hình 13 - Thuật toán Naive Bayes

Bước 12: Hiển thị kết quả sau khi áp dụng thuật toán vào các tập train/test

A screenshot of a computer

Description automatically generated

Hình 14 - Kết quả của quá trình Train/Test.

**3 – Tuần 3: Xử lý dữ liệu cần phân loại gắn nhãn.**

Trong tuần 3, em đã gắn nhãn và xử lý các loại dữ liệu cần sử dụng.

#### Bước 1: Cài đặt thư viện

A screenshot of a computer

Description automatically generated

Hình 15 - Các thư viện.

#### Bước 2: Khai báo và sử dụng các class.

A black screen with white text

Description automatically generated

Hình 16 - Khai báo class cần được sử dụng.

#### Bước 3: Khai báo các từ ngữ tiếng việt chứa dấu

A computer screen with white text

Description automatically generated

Hình 17 - Khai báo các âm tiếng việt có dấu.

#### Bước 4: Hàm Unicode dựng sẵn về Unicode tổ hợp phù hợp tiếng Việt

A screen shot of a computer

Description automatically generated

Hình 18 - Chuyển về Unicode chuẩn.

#### Bước 5: Khai báo nguyên ấm trong tiếng Việt và chuẩn dấu.

A screenshot of a computer

Description automatically generated

Hình 19 - Nguyên và dấu của chúng tiếng Việt.

#### Bước 6: Kiểm tra xem có phải tiếng Việt.

A computer screen with white text and colorful text

Description automatically generated

Hình 20 - Check tiếng Việt.

#### Bước 7: Chuẩn hóa dấu câu tiếng Việt

A computer screen with white and orange text

Description automatically generated

Hình 21 - Chuẩn hóa dấu câu trong tiếng Việt.

#### Bước 8: Xác định các hàm xử lý chuỗi nhập vào.

A computer screen shot of a program code

Description automatically generated

Hình 22 - Hàm Main xử lý chuỗi

#### Bước 9: Xây dựng sự lựa chọn nhập dữ liệu từ bàn phím

A black screen with white text

Description automatically generated

Hình 23 - Nhập dữ liệu từ bàn phím

#### Bước 10: Xây dựng upload dữ liệu

A computer screen with text and images

Description automatically generated

Hình 24 - Upload dữ liệu

#### Bước 11: Tạo sự lựa chọn giữa việc nhập và upload dữ liệu cần được phân loại

A screen shot of a computer program

Description automatically generated

Hình 25 - Lựa chọn cách phân loại văn bản Insign

#### Bước 12: Nhập dữ liệu cần phân loại gắn nhãn văn bản.

A screen shot of a computer program

Description automatically generated

Hình 26 - Phân loại văn bản

**4 – Tuần 4: Tiến hành làm báo cáo, powerpoint và chỉnh sửa lại các lỗi sai trong code**

Trong tuần 4, em đã tiến hành rà soát lại các lỗi sai nhỏ nhặt trong code và chỉnh sửa lại báo cáo theo lời khuyên của thầy. Ngoài ra, em còn cố gắng tối ưu thêm sao cho code chạy ít có lỗi sai nhất.

Thêm vào đó, hoàn thành báo cáo này và tiến hành xây dựng power point cho thuyết trình cũng là một mục tiêu cần phải làm để chuẩn bị cho buổi báo cáo đồ án.

1. **KẾT QUẢ ĐẠT ĐƯỢC**

Những kết quả đạt được trong đồ án này là em có thể phân loại được những chủ đề của bài báo. Giúp cho các người đăng dễ dàng phân loại được chủ đề của bài báo mà họ cần đăng.

Ngoài ra cũng giúp tăng hiệu suất làm việc cũng như tối ưu hóa trải nghiệm người dùng hơn.

1. **TÀI LIỆU THAM KHẢO**

Vương Quốc Bình - Giảng Viên Đại học Công Nghiệp Hà Nội

Github: <https://github.com/binhvq>

Cong Duy Vu Hoang nhà nghiên cứu ứng dụng tải Oracle

Github: <https://github.com/duyvuleo>

Website Python.

Website: [Python Tutorial (w3schools.com)](https://www.w3schools.com/python/default.asp)

Thư viện Underthesea

Link: <http://undertheseanlp.com>

Thuật toán Naive Bayes

Wiki: [Định lý Bayes – Wikipedia tiếng Việt](https://vi.wikipedia.org/wiki/%C4%90%E1%BB%8Bnh_l%C3%BD_Bayes)

Model học máy Naive Bayes

Link: [sklearn.naive\_bayes.MultinomialNB — scikit-learn 1.0.1 documentation](https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html)

Thư viện flask

Link: <https://flask.palletsprojects.com/en/3.0.x/>

Thư viện sklearn

Link: <https://scikit-learn.org/>

1. **DEMO KẾT QUẢ ĐỒ ÁN**

Sau khi xây dựng xong chương trình em sẽ tiến hành kiểm thử sản phẩm. Sau đây em sẽ cho nội dung của bài báo vào một file gọi tạm là test.txt và chạy nó trên chương trình

A screenshot of a computer

Description automatically generated

*Hình 27 – Nhập file test.txt vào chương trình để tiến hành phân loại bài báo*

Chương trình xuất ra nhãn tương ứng với chủ đề của bài báo

A screenshot of a computer

Description automatically generated

*Hình 28 – Chương trình xuất ra chủ đề của bài báo*

Link github đồ án của em: https://github.com/ApplePl3/PYTHON-PROJECT

1. **DOCSTRING**

#@title Giải nén GG Drive

#@title Giải nén file datadata

#@title Thống kê số lượng data theo nhãn

#@title Thống kê các word xuất hiện ở tất cả các nhãn

#@title loại stopword khỏi dữ liệu + lưu file dùng về sau

#@title Chia tập train/test

# Lưu train/test data

# Giữ nguyên train/test để về sau so sánh các mô hình cho công bằng

# encode label

#@title Thêm đường dẫn cho Model

#@title Thuật toán Naive Bayes

# https://scikit-learn.org/stable/modules/generated/sklearn.naive\_bayes.MultinomialNB.html

# Save model

#@title  Xem kết quả trên từng nhãn

#@title Cài đặt thư viện

#@title Khai báo thư viện

#@title Khai báo các âm có dấu trong tiếng việt

# Hàm chuyển Unicode dựng sẵn về Unicde tổ hợp (phổ biến hơn)

#@title Hàm main xử lý chuỗi nạp vào

    # xóa html code

# chuẩn hóa unicode

# chuẩn hóa cách gõ dấu tiếng Việt

 # tách từ

# đưa về lower

# xóa các ký tự không cần thiết

# xóa khoảng trắng thừa

#Nhập dữ liệu từ bàn phím

#Dữ liệu được tải lên

#@title Nhập sự lựa chọn của bạnbạn

#@title Nạp dữ liệu cần phân loại

#document = input("Nhập dữ liệu cần xem xét: ")