

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
ĐẠI HỌC KINH TẾ TP HỒ CHÍ MINH  
TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ



## ĐỒ ÁN MÔN HỌC

### ĐỀ TÀI:

CRAWL DỮ LIỆU BÁO ĐIỆN TỬ CÓ LIÊN QUAN ĐẾN  
**DONALD TRUMP** TỪ **ALLSIDES.COM**, THỰC HIỆN PHÂN  
TÍCH CẢM XÚC BÀI BÁO VÀ HÌNH ẢNH LIÊN QUAN. XÂY  
DỰNG CHỨC NĂNG TÌM KIẾM NỘI DUNG DỰA TRÊN HÌNH  
ẢNH

Học phần: Big Data & Application

Nhóm Sinh Viên:

1. Phan Trần Sơn Bảo
2. Nguyễn Phúc Hải
3. Nguyễn Đình Đại Nhon

Chuyên Ngành: KHOA HỌC DỮ LIỆU

Khóa: K46

Giảng Viên: TS. Đặng Nhân Cách

TP. Hồ Chí Minh, Ngày 04 tháng 04 năm 2023

# MỤC LỤC

|   |                                     |
|---|-------------------------------------|
| <b>MỤC LỤC .....</b>  | <b>1</b>                            |
| <b>LỜI MỞ ĐẦU .....</b>   | <b>3</b>                            |
| <b>NHẬN XÉT GIẢNG VIÊN.....</b>   | <b>Error! Bookmark not defined.</b> |
| <b>CHƯƠNG 1. TỔNG QUAN .....</b>  | <b>4</b>                            |
| 1.1 GIỚI THIỆU VỀ ĐỀ TÀI .....  | 4                                   |
| 1.2. PHÁT BIỂU BÀI TOÁN.....  | 4                                   |
| 1.3. MỘT SỐ HƯỚNG TIẾP CẬN GIẢI QUYẾT BÀI TOÁN .....  | 5                                   |
| 1.4. Các công cụ và thư viện cần dùng. ....   | 5                                   |
| <b>CHƯƠNG 2. CƠ SỞ LÝ THUYẾT .....</b>  | <b>9</b>                            |
| 2.1. MÔ HÌNH HỌC MÁY.....   | 9                                   |
| 2.1.1. TEXT SENTIMENT ANALYSIS (PHÂN TÍCH CẢM XÚC VĂN BẢN).....   | 9                                   |
| 2.2. MÔ HÌNH HỌC SÂU.....   | 10                                  |
| 2.2.1. CBIR (TÌM KIẾM HÌNH ẢNH DỰA TRÊN NỘI DUNG – CONTENT-BASED IMAGE RETRIEVAL).....                    | 10                                  |
| 2.2.2. IMAGE SENTIMENT ANALYS (PHÂN TÍCH CẢM XÚC HÌNH ẢNH).....   | 11                                  |
| <b>CHƯƠNG 3. CRAWL DATA .....</b>   | <b>12</b>                           |
| 3.1. CRAWL SỐ PAGE.....   | 12                                  |
| 3.2. CRAWL BẢNG DATA VỚI MỘT SỐ FEATURE.....  | 13                                  |
| 3.3. CRAWL SOURCE LINK.....   | 17                                  |
| 3.4. CRAWL IMAGE.....   | 18                                  |
| 3.5. CRAWL TEXT .....   | 19                                  |
| <b>CHƯƠNG 4. XỬ LÝ VĂN BẢN , PHÂN TÍCH MỨC ĐỘ CẢM XÚC &amp; ỨNG DỤNG CONTENT-BASED IMAGE SEARCH .....</b> | <b>24</b>                           |
| 4.1. LÀM SẠCH VĂN BẢN.....  | 24                                  |
| 4.2. ĐÁNH GIÁ CẢM XÚC VĂN BẢN (SENTIMENT ANALYSIS) .....  | 28                                  |
| Hình 4.2.1. Đánh giá sentiment_score() và magnitude_score() .....   | 29                                  |
| 4.3. ỨNG DỤNG TÌM KIẾM NỘI DUNG DỰA TRÊN HÌNH ẢNH (CBIR – CONTENT-BASED IMAGE RETRIEVAL) .....            | 32                                  |
| 4.4. ĐÁNH GIÁ CẢM XÚC HÌNH ẢNH .....  | 36                                  |
| <b>CHƯƠNG 5. KẾT LUẬN .....</b>   | <b>44</b>                           |
| 5.1. CÁC KẾT QUẢ ĐẠT ĐƯỢC .....   | 44                                  |
| 5.2. NHỮNG HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN .....  | 44                                  |
| <b>TÀI LIỆU THAM KHẢO VÀ CÔNG CỤ HỖ TRỢ.....</b>  | <b>45</b>                           |

|                      |                                     |
|----------------------|-------------------------------------|
| <b>PHỤ LỤC .....</b> | <b>Error! Bookmark not defined.</b> |
|----------------------|-------------------------------------|

## LỜI MỞ ĐẦU

Thời đại hiện nay là thời đại của công nghệ, với sự phát triển một cách chóng mặt về các công nghệ, các đổi mới về mặt thiết bị đã ảnh hưởng đến xu hướng đọc báo và tin tức của xã hội hiện nay. Ngoài các trang báo điện tử, nhóm nhận thấy rằng đang có một xu hướng phát triển trong việc sử dụng mạng xã hội để cập nhật tin tức. Và với một số sự kiện về việc đăng tin giả, nhóm đã quyết định chọn đề tài “Xây dựng chương trình crawl dữ liệu báo điện tử có liên quan đến donald trump từ Allsides.com & sử dụng dữ liệu crawl phục vụ cho mục đích xây dựng hệ thống tìm kiếm nội dung dựa trên hình ảnh, phân tích cảm xúc báo điện tử dựa trên cả nội dung văn bản và nội dung hình ảnh” để làm đồ án kết thúc học phần Dữ liệu lớn và ứng dụng.

Trong quá trình thực hiện đồ án, nhóm có thể có các sai sót về mặt lý thuyết cũng như mặt kỹ thuật thực hiện. Nhóm em mong rằng sẽ nhận được các nhận xét từ thầy để nhóm có thể cải thiện thêm trong tương lai.

Thay mặt nhóm sinh viên thực hiện đồ án, em xin phép gửi lời cảm ơn đến thầy Đặng Nhân Cách vì đã có sự hỗ trợ, hướng dẫn trong quá trình học tập cũng như quá trình thực hiện đồ án. Sự hỗ trợ, giảng dạy của thầy là nhân tố rất lớn góp phần giúp cho nhóm sinh viên chúng em hoàn thiện được đồ án kết thúc học phần này. Một lần nữa, em xin phép được gửi lời cảm ơn chân thành nhất đến thầy.

Thay mặt nhóm sinh viên thực hiện đồ án

Sinh viên,

Hải

Nguyễn Phúc Hải

# CHƯƠNG 1. TỔNG QUAN

## 1.1 GIỚI THIỆU VỀ ĐỀ TÀI

Với sự thay đổi về mặt công nghệ hiện nay, đặc biệt là sau mùa dịch, các thiết bị di động, laptop, máy tính bảng,... được càng nhiều người sử dụng hơn vì sự tiện lợi, nhanh chóng và hiện đại, hợp thời của chúng. Và với việc các thiết bị điện tử được sử dụng nhiều hơn, số lượng người sử dụng mạng xã hội, đọc báo điện tử cũng tăng lên một cách chóng mặt. Ngày nay, báo giấy không còn được ưa chuộng nhiều như trước. Tuy nhiên, giữa mạng xã hội và đọc báo điện tử, xã hội đang có xu hướng nghiêng về phía bên mạng xã hội hơn. Ta có thể dễ dàng thấy được điều đó thông qua việc mở rộng “địa bàn hoạt động” của các trang báo điện tử. Các trang báo đã bắt đầu mở ra các trang (page) trên mạng xã hội (được đánh dấu “tích xanh” có bản quyền). Bên cạnh các trang báo được sự công nhận của nhà nước, cũng có các trang mạng xã hội đăng tin một cách lẫn lộn, tràn lan, có cả tin giả tin thật khiến nhiều người hoang mang, nhận được các thông tin sai lệch dẫn đến các hiểu lầm nghiêm trọng. Nhận thấy điều này, nhóm đã lên ý tưởng về một hệ thống được tích hợp vào một trang báo tập hợp các bài báo một cách trung lập từ những trang báo lớn có uy tín với chức năng tìm kiếm nội dung, bài báo có hình ảnh giống với hình ảnh được đưa vào để tìm kiếm. Lý do cho việc này là các trang đăng tin tức (kể cả chính thống lẫn không chính thống, tin giả lẫn tin thật) ở trên các trang mạng xã hội thường có đăng kèm hình ảnh, và chúng thường giống nhau. Và với việc cần kiểm tra lại tin tức, người dùng có thể sử dụng hình ảnh đó để tìm kiếm các bài báo có hình ảnh liên quan để kiểm chứng lại thông tin do các trang mạng xã hội đăng tải.

Ngoài ra, như đã đề cập bên trên, các bài báo, các tin tức khi được đăng sẽ đi kèm với hình ảnh. Việc phân tích cảm xúc văn bản và hình ảnh sẽ cần thiết khi cần đánh giá mức độ cảm xúc đối với các bài báo, tin tức vì các hình ảnh đi kèm sẽ liên quan đến nội dung bên trong bài báo, tin tức. Việc phân tích cảm xúc hình ảnh có tính trực quan, dễ nhìn dễ hình dung cũng như dễ cảm nhận được cảm xúc. Việc phân loại các bài báo, tin tức có tính tiêu cực, tích cực dựa trên hình ảnh và cả văn bản sẽ có độ chính xác cao hơn. Nếu kết hợp với hệ thống tìm kiếm nội dung dựa trên hình ảnh đã được đề cập ở bên trên. Hệ thống sẽ có thể giúp người đọc biết được rằng tin tức, bài báo mình đang tìm kiếm là tiêu cực hay tích cực.

## 1.2. PHÁT BIỂU BÀI TOÁN

Có 4 bài toán ở đây cần được giải quyết. Để có thể lấy được dữ liệu về để phân tích cảm xúc văn bản, hình ảnh và huấn luyện mô hình tìm kiếm nội dung dựa trên hình ảnh. Ta còn 1 bài toán cần được giải quyết nữa là lấy được dữ liệu từ một trang tin tức chuyên thu thập các bài báo, tin tức một cách khách quan từ các trang báo có uy tín. Ở đây nhóm chọn trang Allsides.com, một trang chuyên thu thập thông tin từ hơn 1400 trang báo lớn nhỏ như NYTimes, ABC News, BBC News, Bloomberg,... trang này có cả phần đánh giá về mức độ xu hướng chính trị của các trang báo thông qua các nghiên cứu, các phương pháp đánh giá riêng của Allsides.com. Việc thu thập dữ liệu sẽ được thực hiện thủ công

dựa trên việc gửi requests đến Allsides.com để lấy các thẻ HTML có chứa các thông tin cần thiết (hình ảnh, tiêu đề, mô tả bài báo, link hình ảnh, bài báo, link dẫn đến link bài báo gốc (của trang báo đăng gốc), bias (xu hướng chính trị)... ) từ những thông tin đó, ta bắt đầu giữ lại các thông tin cần thiết như bias, tên của tờ báo đăng tin, mô tả tổng quan về bài báo,... còn lại sẽ được thay thế bằng nội dung từ bài báo gốc (link hình ảnh, nội dung bài báo,...) và có những xử lý sơ bộ qua bộ dữ liệu. Sau khi đã có được bộ dữ liệu hoàn chỉnh gồm các thông tin về bias, tên tòa báo đăng tin, link hình ảnh, tiêu đề bài báo, mô tả bài báo, nội dung bài báo,... 3 bài toán chính là phân tích cảm xúc văn bản, phân tích cảm xúc hình ảnh và tìm kiếm nội dung dựa trên hình ảnh sẽ được thực hiện.

### **1.3. MỘT SỐ HƯỚNG TIẾP CẬN GIẢI QUYẾT BÀI TOÁN**

#### **1.4. Các công cụ và thư viện cần dùng.**

- Phần 1:
  - BeautifulSoup:
    - BeautifulSoup là một thư viện Python mã nguồn mở được sử dụng để phân tích cú pháp HTML và XML. Thư viện này cho phép bạn dễ dàng trích xuất dữ liệu từ các tài liệu HTML hoặc XML, giúp tăng hiệu quả trong việc lấy dữ liệu từ web và xử lý dữ liệu.
    - Một số tính năng nổi bật của thư viện BeautifulSoup bao gồm:
      - Hỗ trợ phân tích cú pháp HTML và XML.
      - Cho phép truy cập vào các phần tử HTML và XML theo tên, thuộc tính, vị trí, v.v.
      - Cung cấp các phương thức để trích xuất và lấy dữ liệu từ các phần tử HTML và XML, bao gồm cả các phần tử lồng nhau.
      - Cung cấp các phương thức để tìm kiếm các phần tử HTML và XML dựa trên các tiêu chí tìm kiếm, bao gồm cả tên, thuộc tính và văn bản.
      - Hỗ trợ việc tìm kiếm và xử lý các dữ liệu không có cấu trúc rõ ràng.
      - Hỗ trợ một số tính năng nâng cao như phân tích cú pháp các trang web được tạo bằng JavaScript và các thư viện khác để phân tích cú pháp.
    - Thư viện BeautifulSoup là một trong những thư viện phổ biến nhất để phân tích cú pháp HTML và XML trong Python và được sử dụng rộng rãi trong các ứng dụng web và khai thác dữ liệu.
    -
  - Requests:
    - Requests là một thư viện Python mã nguồn mở cho phép gửi các yêu cầu HTTP/1.1 với rất nhiều tính năng tiện lợi. Thư viện này được thiết kế để dễ sử dụng và cho phép truy cập dữ liệu HTTP một cách dễ dàng.
    - Một số tính năng nổi bật của thư viện Requests bao gồm:

- Gửi các yêu cầu HTTP đơn giản hoặc phức tạp, bao gồm cả các yêu cầu POST, GET, PUT, DELETE, HEAD và OPTIONS.
- Hỗ trợ các phương thức chứng thực, bao gồm cả Basic Auth, Digest Auth và OAuth.
- Cho phép gửi các yêu cầu với các header tùy chỉnh, cookie và proxy.
- Tự động quản lý session và giữ các cookie giữa các yêu cầu.
- Hỗ trợ các định dạng dữ liệu phổ biến như JSON và XML.
- Cung cấp giao diện đơn giản để xử lý dữ liệu trả về.
- Thư viện Requests là một trong những thư viện phổ biến nhất để làm việc với HTTP trong Python và được sử dụng rộng rãi trong nhiều ứng dụng web và dịch vụ API.

- Phần 2:

- Newsplease:

- Thư viện NewsPlease là một công cụ mã nguồn mở cho phép bạn thu thập tin tức và bài báo từ các trang web tin tức khác nhau. Thư viện này được viết bằng ngôn ngữ Python và sử dụng kỹ thuật web scraping để lấy dữ liệu từ các trang web tin tức.

- Các tính năng chính của NewsPlease bao gồm: Thu thập tin tức và bài báo từ các trang web tin tức khác nhau, tự động trích xuất các thông tin quan trọng như tiêu đề, nội dung, tác giả, ngày đăng, v.v. Hỗ trợ lưu trữ dữ liệu được lấy về dưới dạng JSON hoặc trong cơ sở dữ liệu MongoDB

- NewsPlease là một thư viện rất hữu ích cho các nhà nghiên cứu, nhà báo và các nhà phát triển muốn xây dựng các ứng dụng liên quan đến việc thu thập dữ liệu tin tức và phân tích tin tức.

- Phần 3:

- AFINN:

Thư viện AFINN là một thư viện phân tích cảm xúc dựa trên danh sách từ. Nó sử dụng một bảng điểm để gán điểm cho các từ trong văn bản. Thư viện này có thể được sử dụng để phân tích cảm xúc của văn bản bằng cách sử dụng phương thức score(). AFINN gán điểm số cảm xúc cho các từ trong khoảng từ -5 đến 5, với điểm số âm cho biết cảm xúc tiêu cực và điểm số dương cho biết cảm xúc tích cực. Tổng điểm của các từ trong văn bản sẽ

được tính tổng các điểm của từng từ (do cách thức tính toán này, điểm số sentiment của nhóm khi tính ra sẽ có con số “tương đối” lớn).

Thư viện AFINN được phát triển bởi Finn Årup Nielsen và có hơn 3300 từ với điểm số liên quan đến mỗi từ. Thư viện này có thể được sử dụng để phân tích cảm xúc của văn bản bằng cách tính tổng điểm của các từ trong văn bản.

- KERAS:

Keras là một thư viện mạng nơ-ron mã nguồn mở được viết bằng Python cung cấp API cấp cao để xây dựng và huấn luyện các mô hình học sâu. Nó được phát triển bởi François Chollet, một kỹ sư của Google, là một phần của dự án ONEIRO (Hệ thống hoạt động Robot thông minh điện tử Neuro-Electronic Open-ended). Keras có khả năng chạy trên TensorFlow, Theano hoặc CNTK. Nó thân thiện với người dùng, có tính mô-đun và có thể mở rộng để tạo điều kiện cho việc thử nghiệm nhanh hơn với các mạng nơ-ron học sâu.

KERAS có thể được sử dụng để xây dựng các mô hình học sâu, bao gồm các mô hình được sử dụng trong các hệ thống truy xuất hình ảnh dựa trên nội dung (CBIR). Các hệ thống CBIR truy xuất hình ảnh dựa trên độ tương tự về thuộc tính và có thể được đánh giá bằng trung bình độ chính xác trung bình (MAP) là trung bình của độ chính xác tại mỗi lần truy cập.

- Pytorch:

- là một thư viện mã nguồn mở và mạnh mẽ cho máy học và học sâu (deep learning) được phát triển bởi Facebook. PyTorch cho phép người dùng xây dựng các mô hình mạng nơ-ron (neural networks) cho các ứng dụng như nhận dạng hình ảnh, xử lý ngôn ngữ tự nhiên (NLP), dự đoán chuỗi thời gian và nhiều ứng dụng khác.

- Các tính năng chính của PyTorch bao gồm: Tính toán đa luồng trên GPU cho việc xử lý tăng tốc tính toán, cung cấp một giao diện trực quan và dễ sử dụng cho việc xây dựng các mô hình mạng nơ-ron, hỗ trợ tính toán gradient tự động (automatic differentiation) để thuận tiện cho việc huấn luyện mô hình, cho phép tạo ra các mô hình dựa trên sự kế thừa (inheritance) hoặc sự tập hợp (composition)

- PyTorch cũng có một cộng đồng lớn và đa dạng, cung cấp nhiều tài liệu, ví dụ và các hướng dẫn về việc sử dụng PyTorch cho các ứng dụng máy học và học sâu. PyTorch được xem là một trong những thư viện phổ biến nhất và mạnh mẽ nhất trong lĩnh vực máy học và học sâu hiện nay.



○ Scikit-learn:

- Scikit-learn là một thư viện mã nguồn mở và phổ biến nhất để thực hiện các tác vụ máy học (machine learning) trong Python. Thư viện này được thiết kế để cung cấp cho người dùng các công cụ và thuật toán phổ biến để xử lý các bài toán phân loại, dự đoán, phân cụm và hồi quy.

- Các tính năng chính của Scikit-learn bao gồm: Các thuật toán học máy phổ biến như phân loại, hồi quy, phân cụm và giảm chiều dữ liệu, các công cụ để tiền xử lý dữ liệu, chọn đặc trưng và đánh giá mô hình, hỗ trợ cho nhiều loại dữ liệu khác nhau, bao gồm dữ liệu số, văn bản và hình ảnh, hỗ trợ cho các kỹ thuật như học máy tự động (automated machine learning), trang bị (ensemble learning) và học có giám sát (supervised learning), các hàm tiện ích cho việc chọn siêu tham số (hyperparameter tuning), cross-validation và huấn luyện mô hình...

- Scikit-learn được sử dụng rộng rãi trong cả nghiên cứu lẫn ứng dụng thực tế trong các lĩnh vực như khoa học dữ liệu, y tế, tài chính, thương mại điện tử và nhiều lĩnh vực khác.

○ NLTK:

Bộ công cụ Ngôn ngữ Tự nhiên (Natural Language Toolkit - NLTK) là một thư viện Python cung cấp các công cụ xử lý và làm sạch dữ liệu văn bản. Nó được sử dụng rộng rãi trong các tác vụ Xử lý Ngôn ngữ Tự nhiên (Natural Language Processing - NLP) như phân tích cảm xúc, phân loại chủ đề và phân loại văn bản. NLTK cung cấp một loạt các công cụ để làm sạch văn bản như loại bỏ dấu câu, chuyển đổi văn bản thành chữ thường, loại bỏ từ dừng, thu gọn từ và lemmatization.

Các chức năng làm sạch văn bản được cung cấp trong thư viện NLTK:

Tokenization - Chia câu và từ từ nội dung của văn bản.

Loại bỏ ký tự đặc biệt và số.

Loại bỏ từ dừng (stopwords) - Từ dừng là các từ không có ý nghĩa trong câu như 'the', 'is', 'a', vv.

Stemming - Giảm từ xuống dạng gốc của chúng như 'running' thành 'run'.

Lemmatization - Giảm từ xuống dạng cơ sở của chúng như 'running' thành 'run' nhưng khác với stemming ở chỗ nó giảm từ xuống dạng từ cơ bản có trong từ điển của từ.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.1. MÔ HÌNH HỌC MÁY

#### 2.1.1. TEXT SENTIMENT ANALYSIS (PHÂN TÍCH CẢM XÚC VĂN BẢN)

Phân tích cảm xúc văn bản (sentiment analysis) là bài toán được nghiên cứu trong lĩnh vực Xử lý ngôn ngữ tự nhiên. Mục tiêu của bài toán là tìm ra cảm xúc (tích cực, tiêu cực, trung tính) của một câu chữ trong một lĩnh vực cụ thể nào đó. Sentiment Analysis là một ứng dụng trí tuệ nhân tạo, nó sử dụng các thuật toán phức tạp để xử lý ngôn ngữ tự nhiên của con người (NLP) và xác định các đặc điểm cảm xúc tiêu cực/tích cực tại một thời điểm thông qua văn bản hoặc lời nói.<sup>1</sup>

Quy trình phân tích cảm xúc văn bản thường được chia thành các bước sau:

1. Thu thập và tiền xử lý dữ liệu: Thu thập và tiền xử lý văn bản để loại bỏ nhiễu và chuẩn bị dữ liệu cho việc phân tích cảm xúc. Các bước tiền xử lý bao gồm tách từ, loại bỏ stop words, chuyển đổi các từ về dạng chuẩn và loại bỏ các ký tự đặc biệt.
2. Xây dựng bộ từ điển: Xây dựng bộ từ điển chứa các từ và cụm từ liên quan đến các cảm xúc cần phân tích.
3. Phân tích cảm xúc: Sử dụng các phương pháp khác nhau để phân tích cảm xúc của văn bản, bao gồm:

Phân loại học máy: Sử dụng các thuật toán học máy, chẳng hạn như Naive Bayes, Support Vector Machine (SVM), Random Forest để phân loại cảm xúc của văn bản.

Phân tích dựa trên quy tắc: Sử dụng các quy tắc để xác định cảm xúc của văn bản. Các quy tắc này có thể được xây dựng dựa trên kinh nghiệm hoặc các nguồn tài liệu liên quan.

Phân tích dựa trên định lượng: Sử dụng các phương pháp định lượng để xác định mức độ cảm xúc của văn bản. Các phương pháp này bao gồm đếm số lần xuất hiện của các từ và cụm từ liên quan đến các cảm xúc cần phân tích.

4. Đánh giá mô hình: Đánh giá hiệu quả của mô hình phân tích cảm xúc bằng cách sử dụng các phương pháp đánh giá mô hình, chẳng hạn như precision, recall và F1-score.
5. Triển khai mô hình: Triển khai mô hình phân tích cảm xúc vào ứng dụng thực tế để phân tích cảm xúc của văn bản.

Các kỹ thuật phân tích cảm xúc có thể được phân loại thành các phương pháp học máy, phương pháp dựa trên từ điển và thậm chí là các phương pháp kết hợp. Một số phân loại nghiên cứu khác trong phân tích cảm xúc bao gồm: phân tích cảm xúc đa phương tiện,

phân tích cảm xúc dựa trên khía cạnh, phân tích ý kiến tình vi và phân tích cảm xúc theo ngôn ngữ cụ thể.

Gần đây, các kỹ thuật học sâu như RoBERTa và T5 được sử dụng để huấn luyện các bộ phân loại cảm xúc hiệu suất cao, được đánh giá bằng các chỉ số như F1, recall và precision. Để đánh giá hệ thống phân tích cảm xúc, các tập dữ liệu thử nghiệm như SST, GLUE và đánh giá phim IMDB được sử dụng.

## **2.2. MÔ HÌNH HỌC SÂU**

### **2.2.1. CBIR (TÌM KIẾM HÌNH ẢNH DỰA TRÊN NỘI DUNG – CONTENT-BASED IMAGE RETRIEVAL)**

Tìm kiếm hình ảnh dựa trên nội dung (Content-Based Image Retrieval, CBIR) là một kỹ thuật được sử dụng để tìm kiếm và truy xuất hình ảnh từ cơ sở dữ liệu lớn dựa trên nội dung hình ảnh của chúng. Không giống như các công cụ tìm kiếm dựa trên văn bản truyền thống dựa vào các mô tả văn bản hoặc siêu dữ liệu để truy xuất hình ảnh, CBIR dựa vào các đặc điểm trực quan của hình ảnh, chẳng hạn như màu sắc, họa tiết, hình dạng và bố cục không gian.

Quá trình thực hiện kỹ thuật tìm kiếm này thường có 4 giai đoạn:

- a. Biểu diễn hình ảnh (Image presentation): hình ảnh sẽ được chuyển đổi thành một tập hợp các thuộc tính trực quan có thể được sử dụng để tìm kiếm tương đồng và truy xuất. Các thuộc tính phổ biến được sử dụng trong CBIR bao gồm biểu đồ màu sắc, mô tả họa tiết, độ lớn hình ảnh và các thuộc tính có ý nghĩa đối với mạng nơ ron tích chập sâu.
- b. Xếp chỉ mục (Indexing): sau khi có được thuộc tính từ các hình ảnh, cơ chế indexing sẽ sắp xếp các thuộc tính này vào cơ sở dữ liệu có thể truy xuất/tìm kiếm như phân cụm, hashing (hàm băm) và tree-based indexing
- c. Xử lý truy vấn (Query processing): Khi người dùng đưa vào một hình ảnh truy vấn, các thuộc tính của hình ảnh đó sẽ được so sánh với các hình ảnh trong cơ sở dữ liệu để truy xuất các hình ảnh giống nhau nhất (về mặt thuộc tính). So sánh này có thể được thực hiện bằng cách sử dụng nhiều phép đo tương đồng: khoảng cách Euclidean, độ tương đồng cosine hoặc độ đo tương quan (correlation).
- d. Ý kiến phản hồi (Relevance feedback): Sau lần truy xuất ban đầu, mức độ liên quan của các hình ảnh đã truy xuất có thể được sử dụng để tinh chỉnh các tham số trong thuật toán và sử dụng kết quả của các lần truy xuất trước đó để thay đổi dữ liệu đầu vào trong bước huấn luyện mô hình nhằm cải thiện độ chính xác của kết quả truy xuất ở những lần sau.

### **2.2.2. IMAGE SENTIMENT ANALYSIS (PHÂN TÍCH CẢM XÚC HÌNH ẢNH)**

Image sentiment analysis là một tác vụ trong xử lý ngôn ngữ tự nhiên và thị giác máy tính, mục đích là phân tích cảm xúc được thể hiện trong hình ảnh. Tác vụ này nhằm đánh giá tính chất của hình ảnh, đưa ra mô tả về nội dung và cảm xúc được thể hiện trong đó, giúp các doanh nghiệp, tổ chức hiểu hơn về sự phản hồi của khách hàng đối với sản phẩm, dịch vụ hoặc thương hiệu của họ.

Các phương pháp trong image sentiment analysis có thể chia thành hai loại chính là phương pháp dựa trên đặc trưng và phương pháp học sâu. Phương pháp dựa trên đặc trưng thường sử dụng các kỹ thuật trích xuất đặc trưng từ hình ảnh, như màu sắc, hình dạng và kích thước để xác định cảm xúc của hình ảnh. Phương pháp học sâu sử dụng các kiến trúc mạng nơ-ron sâu để học và phân tích các đặc trưng của hình ảnh, đưa ra dự đoán về cảm xúc của hình ảnh.

Quy trình trong Image Sentiment Analysis bao gồm các bước sau:

1. Thu thập dữ liệu: Thu thập các hình ảnh từ các nguồn khác nhau để xây dựng tập dữ liệu đủ lớn và đa dạng để đào tạo mô hình phân tích cảm xúc.
2. Tiền xử lý: Tiền xử lý hình ảnh để chuẩn bị dữ liệu cho việc phân tích cảm xúc. Các bước tiền xử lý bao gồm chuyển đổi hình ảnh thành dạng số, cắt ảnh để loại bỏ nhiễu và giảm kích thước hình ảnh để giảm thời gian xử lý.
3. Trích xuất đặc trưng: Trích xuất các đặc trưng từ hình ảnh bằng cách sử dụng các phương pháp khác nhau, chẳng hạn như phân tích màu sắc, phân tích hình dạng, phân tích kết cấu.
4. Xây dựng mô hình: Xây dựng một mô hình phân tích cảm xúc từ dữ liệu huấn luyện bằng cách sử dụng các phương pháp học máy hoặc học sâu. Các mô hình phổ biến trong Image Sentiment Analysis bao gồm Convolutional Neural Networks (CNN) và Recurrent Neural Networks (RNN).
5. Huấn luyện mô hình: Huấn luyện mô hình phân tích cảm xúc bằng cách sử dụng dữ liệu đào tạo. Việc huấn luyện mô hình có thể mất nhiều thời gian và tài nguyên tính toán, tùy thuộc vào quy mô của dữ liệu huấn luyện và phức tạp của mô hình.
6. Đánh giá mô hình: Đánh giá mô hình phân tích cảm xúc bằng cách sử dụng các phương pháp đánh giá mô hình, chẳng hạn như precision, recall và F1-score.
7. Triển khai mô hình: Triển khai mô hình phân tích cảm xúc vào ứng dụng thực tế để phân tích cảm xúc của hình ảnh.

Các ứng dụng của image sentiment analysis rất đa dạng, bao gồm nhận dạng cảm xúc trong hình ảnh của khách hàng, phân tích đánh giá sản phẩm, xác định sự quan tâm và phản ứng của người dùng đối với quảng cáo trên mạng xã hội, phát hiện các nội dung không phù hợp hoặc vi phạm bản quyền trong hình ảnh, và nhiều ứng dụng khác.

## CHƯƠNG 3. CRAWL DATA

### 3.1. CRAWL SỐ PAGE

```
# creates a empty list to store the story pages from AllSides.com
pages = []

# We only want to extract stories about immigration
story = 'donald-trump'

def get_seed(n):
    """
    n defines the number of pages back to pull
    n=1 steps back to April 2018 (as of April 2020)
    """

    for i in range(0, n+1):
        url = 'https://www.allsides.com/blog/tags/' + \
            str(story) + '?page=' + str(i)
        pages.append(url)

get_seed(9)

pages
```

Tạo hàm `get_seed` trong đó sử dụng vòng lặp `for` chạy từ 0 đến số trang yêu cầu `n`, sử dụng đường dẫn `'https://www.allsides.com/blog/tags/'` kèm thêm chủ đề `donald trump` sau đó thêm hậu tố `?page` để tạo ra các đường dẫn đến các trang chứa link bài viết.

Facebook Twitter Instagram RSS
Donate
Join
User icon
Menu icon

[NEWS](#)
[MEDIA BIAS](#)
[MISINFORMATION](#)
[PERSPECTIVES](#)
[TOPICS](#)
[SERVICES](#)
[ABOUT](#)

[Perspectives Blog](#)
[All Posts](#)
[AllStances](#)
[Bridging Divides](#)
[Fact Check](#)
[Media Bias](#)
[Polarization](#)
[Story of the W](#)

## Tag: Donald Trump

### Is Biden Less Available to the Media than Past Presidents?

FACTS & DATA / MARCH 30TH, 2023 / BY HENRY A. BRECHTER

Biden holds far fewer news conferences than many of his predecessors.

[Read more](#)

21 Shares

Joe Biden, Politics, White House, Donald Trump, Barack Obama, Media Industry

---

### When (or if) a Former President Gets Indicted

RECOMMENDED READING / MARCH 27TH, 2023 / BY DAN SCHNUR

Kết quả sau khi chạy ta được list các link page

```
[ 'https://www.allsides.com/blog/tags/donald-trump?page=0',
  'https://www.allsides.com/blog/tags/donald-trump?page=1',
  'https://www.allsides.com/blog/tags/donald-trump?page=2',
  'https://www.allsides.com/blog/tags/donald-trump?page=3',
  'https://www.allsides.com/blog/tags/donald-trump?page=4',
  'https://www.allsides.com/blog/tags/donald-trump?page=5',
  'https://www.allsides.com/blog/tags/donald-trump?page=6',
  'https://www.allsides.com/blog/tags/donald-trump?page=7',
  'https://www.allsides.com/blog/tags/donald-trump?page=8',
  'https://www.allsides.com/blog/tags/donald-trump?page=9' ]
```

### 3.2. CRAWL BẢNG DATA VỚI MỘT SỐ FEATURE

```
def soup_basics(item):
    # Send a request to the URL and get the response
    response = requests.get(item)

    # Parse the HTML content of the response using BeautifulSoup
    soup = BeautifulSoup(response.content, 'html.parser')

    return soup
```

Tạo hàm `soup_basics` sử dụng thư viện `request` gửi yêu cầu tới url được chỉ định sau đó phân tích nội dung HTML được phản hồi từ url.

```
def harvest_links(pages):
    """
    runs the parser over submitted pages
    identifies headline link content in the extracted page
    appends relevant links to a list
    """
    for item in pages:
        soup=soup_basics(item)

        # Find all the links on the page that point to blog posts
        blog_links = []
        for link in soup.find_all('a'):
            href = link.get('href')
            if href and '/blog/' in href:
                blog_links.append(href)

        # Print the list of blog links
        # print(blog_links)
        for i in blog_links:
            if('https://www.allsides.com/' in i) : link = i
            else: link= 'https://www.allsides.com/' +i
            if not '/tags/' in link:
                link_harvest.append(link)

    harvest_links(pages)

# story_headline_list
unique_link_harvest= list(set(link_harvest))
unique_link_harvest
# link_harvest
```

Tạo hàm `harvest_links` trong đó sử dụng vòng `for` duyệt tất cả các link page được cào trước đó , gọi hàm `soup_basics` cho các link page , kết quả trả về

```
▼ <div class="view-content">
  ▶ <div class="views-row views-row-1 views-row-odd views-row-first"> ...
    </div> == $0
  ▶ <div class="views-row views-row-2 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-3 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-4 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-5 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-6 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-7 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-8 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-9 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-10 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-11 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-12 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-13 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-14 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-15 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-16 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-17 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-18 views-row-even"> ... </div>
  ▶ <div class="views-row views-row-19 views-row-odd"> ... </div>
  ▶ <div class="views-row views-row-20 views-row-even views-row-last"> ... </div>
```



Với từng các thẻ class ta tham chiếu các thẻ a href để lấy nội dung trong href , đó là link các bài viết trong page.

```
▶ <div class="span4"> ... </div>
▼ <div class="span8">
  ▶ <div class="row-fluid field field-name-body field-type-text-with-summary field-label-hidden clearfix"> ...
  </div>
...
  <a href="/../blog/biden-less-available-media-past-presidents" class="form-submit newreadmore btn btn-success btn-small">Read more</a> == $0
</div>
::after
```

Sau khi chạy xong vòng lặp ta có kết quả đạt được :

```
[ 'https://www.allsides.com/.../blog/trump-sues-cnn-defamation',
  'https://www.allsides.com/blog/online-news-sites-announce-trump-victory-excitement-or-horror',
  'https://www.allsides.com/blog/trump-odds-puerto-rico',
  'https://www.allsides.com/blog/when-or-if-former-president-gets-indicted',
  'https://www.allsides.com/blog/trump-and-north-korea',
  'https://www.allsides.com/.../blog/media-react',
  'https://www.allsides.com/blog/remaking-presidency-trump-cabinet',
  'https://www.allsides.com/blog/real-change-requires-real-voter-turnout',
  'https://www.allsides.com/.../blog/whos-winning-2016',
  'https://www.allsides.com/blog/when-college-football-games-affect-presidential-election',
  'https://www.allsides.com/.../blog/2020-election-results-all-sides-live-blog',
  'https://www.allsides.com/blog/when-republicans-and-democrats-live-alternate-universes',
  'https://www.allsides.com/.../blog/when-united-states-turns-away-world',
  'https://www.allsides.com/blog/story-week-4-views-donald-trump',
  'https://www.allsides.com/.../blog/when-mike-pence-goes-rogue',
  'https://www.allsides.com/blog/when-donald-trump-thinks-he-has-been-cancelled',
  'https://www.allsides.com/blog/trump%E2%80%99s-economic-plan-other-media-constrasts-week',
  'https://www.allsides.com/blog/trump-increases-tariffs-chinese-goods',
  'https://www.allsides.com/.../blog/horowitz-testifies-probe-fbis-trump-investigation',
  'https://www.allsides.com/blog/trump-vs-biden-environmental-policy',
  'https://www.allsides.com/.../blog/story-week-gop-convention',
  'https://www.allsides.com/.../blog/us-and-canada-reach-new-trade-pact-supersede-nafta',
  'https://www.allsides.com/.../blog/trump-vs-biden-education-policy-explained-two-minutes',
  'https://www.allsides.com/.../blog/trump-says-us-will-remain-steadfast-partner-saudi-arabia-foolish-or-strategic',
  'https://www.allsides.com/blog/why-news-outlets-differ-calling-arizona-biden',
```

Sau khi có được tất cả các link từ các trang , ta sử dụng hàm extract\_articles để truy cập vào các link . Gọi hàm soup\_basics cho các link rồi truy cập vào class blog-content-wrapper rồi tham chiếu vào các thẻ a href ta có được những link bài viết chính thống (việc chọn những link có định dạng /news/ là vì những bài báo này đã nghiêng hẳn về phe phái cụ thể ( cánh tả , cánh hữu hay trung lập ))

```
# get all news article links
all_articles = []

def extract_articles(link):
    for link_content in link:
        soup = soup_basics(link_content)

        # locate relevant information within the extracted page
        substory_list = soup.find_all(class_='blog-content-wrapper')

        # loop through the different news sources within each major news story
        if 'Snippets' in str(substory_list):
            for i in range(0, len(substory_list)):
                substory_items = substory_list[i].find_all('a')
                for substory_headline in substory_items:
                    link = substory_headline.get('href')
                    if link and '/news/20' in link:
                        all_articles.append(link)

extract_articles(link_harvest)
unique_all_articles= list(set(all_articles))
unique_all_articles
unique_all_articles[0:99]
```



Với kết quả trả về là list `all_articles` gồm các link , ta ép về kiểu set rồi gọi list bao trọn để những link là duy nhất đặt tên list là `unique_all_articles`

Khởi chạy hàm `extract_content` để tạo ra bảng tạm thời với các cột `date` , `description` , `source_name` , `source_bias` , `source_headline` , `source_link`

Trong đó ta duyệt tất cả link từ `unique_link_articles` , gọi hàm `soup_basics` cho các link , với kết quả trả về , ta tiến hành truy cập vào các thẻ để khai thác thông tin rồi điền vào cột thích hợp như thẻ `update_time` điền vào cột `date` , thẻ `description` điền vào cột `description` , thẻ `url` điền vào cột `source headline` , thẻ `strong` điền vào cột `source bias` , cột `title` điền vào cột `source name` ( có một số thẻ giữa các trang web không giống nhau như `description` nên ta gọi kết quả trả về html là hàm `soup_basic` , hàm này tương tự hàm `soup_basics` chỉ là kết quả trả về là text)

```
import pandas as pd

socket.socket

def csv_encoder(text_string):
    coded = text_string.encode("utf-8").strip()
    return coded

# extract all content
def extract_content(link_harvest):
    """
    for each story, pulls the shared news headline, date, and summary description
    for each news source, identifies the source bias (liberal, conservative, center) & outgoing link
    uses re and .contents to clean harvested text
    writes collected, cleaned data to csv
    """

    # open csv file to store info
    file = open('alldiscontent-f.csv', 'w', newline='', encoding='utf-8')
    filewriter = csv.writer(file)
    filewriter.writerow(['Date', 'Description', 'Source_Name',
                        'Source_Bias', 'Source_Headline', 'Source_Link'])

    try:
        for i in range(len(unique_all_articles)):
            soup = soup_basics(unique_all_articles[i])

            story_date = soup.find(property="og:updated_time")
            parts = str(story_date).split('')
            if len(parts)>1:
                clean_date = parts[1]
            else:
                clean_date = None

            lst_line=[]
            lines = str(soup).split("\n")
            for line in lines:
                if 'description' in line:
                    lst_line.append(line)
            try:
                if len(lst_line[0].split(''))>2:
                    if 'og:description' in lst_line[0].split('') : clean_description=lst_line[0].split('')[0]
                    else : clean_description=lst_line[0].split('')[1]
            except: continue

            substory_source = soup.find(property="og:url")
            parts = str(substory_source).split('')
            if len(parts) > 1 :
                clean_source = parts[1]
```

```

substory_source = soup.find(property='og:url')
parts = str(substory_source).split('')
if len(parts) > 1 :
    clean_source = parts[1]
else:
    clean_source = None
clean_source

soup = soup_basic(all_articles[i])
lst_line=[]
lines = str(soup).split("\n")
for line in lines:
    if 'field-content">From The' in line :
        lst_line.append(line)
clean_bias = find_bias(lst_line)
clean_bias

substory_list = soup.find(property='og:title')
parts = str(substory_list).split('')
if len(parts) > 1 :
    clean_list = parts[1]
else:
    clean_list = None

fileWriter.writerow(
    [clean_date, clean_description, clean_list, clean_bias, clean_source, unique_all_articles[i]])

except socket.error as err:
    print('socket connection error... waiting 10 seconds to retry.')
    del self.sock
    time.sleep(10)
    try_count += 1

file.close()

# running the function
extract_content(unique_all_articles)

```

Kết quả trả về sau khi gọi hàm soup\_basics cho mỗi link :

```

<link rel="short-link" href="https://www.allsides.com/node/195554">
<meta property="og:site_name" content="AllSides">
<meta property="og:type" content="article">
<meta property="og:url" content="https://www.allsides.com/blog/did-trump-obstruct-justice-mar-lago">
<meta property="og:title" content="Did Trump Obstruct Justice at Mar-a-Lago?">
<meta property="og:description" content="Interested in getting next week's story and other AllSides newsle
tters in your">
<meta property="og:updated_time" content="2022-09-01T12:29:21-07:00">
<meta property="og:image" content="https://www.allsides.com/sites/default/files/trump-doj-response-574_0.j
pg">
<meta property="og:image:url" content="https://www.allsides.com/sites/default/files/trump-doj-response-574
_0.jpg">
<meta property="og:image:secure_url" content="https://www.allsides.com/sites/default/files/trump-doj-respo
nse-574_0.jpg">

```

Chạy hàm extract\_content với những link vừa được lấy ở trên ta có được bảng data:

|   | Date                      | Description                                       | Source_Name                                       | Source_Bias | Source_Headline                                   | Source_Link                                       |
|---|---------------------------|---|---|-------------|---|---|
| 0 | 2018-03-08T15:34:58-08:00 | The people who like the sort of tariffs that P... | OPINION: Trump's risky call for protests          | Left        | https://www.allsides.com/news/2018-03-08-1534/... | https://www.allsides.com/news/2018-03-08-1534/... |
| 1 | 2020-05-28T07:02:58-07:00 | President Donald Trump is preparing to sign an... | OPINION: Trump's risky call for protests          | Left        | https://www.allsides.com/news/2020-05-28-0702/... | https://www.allsides.com/news/2020-05-28-0702/... |
| 2 | 2019-12-12T06:52:09-08:00 | The Justice Department's top watchdog on Wedne... | OPINION: Donald Trump has committed a lot of s... | Lean Left   | https://www.allsides.com/news/2019-12-12-0651/... | https://www.allsides.com/news/2019-12-12-0651/... |
| 3 | 2018-07-12T12:49:58-07:00 | President Trump kept everyone guessing to the ... | Trump Indictment Could Be a 2024 Cash Cow         | Center      | https://www.allsides.com/news/2018-07-12-1249/... | https://www.allsides.com/news/2018-07-12-1249/... |
| 4 | 2019-10-17T06:20:50-07:00 | As President Donald Trump continues to fill hi... | Police surround NY Courthouse and DC Capitol      | Center      | https://www.allsides.com/news/2019-10-17-0620/... | https://www.allsides.com/news/2019-10-17-0620/... |

### 3.3. CRAWL SOURCE LINK

```

# get all news article links
all_articles_f = []
def extract_articles(link):
    for i in range(len(d['Source_Headline'])):
        soup = soup_basics(d['Source_Headline'][i])

        # locate relevant information within the extracted page
        substory_list = soup.find_all(class_='read-more-story')

        # loop through the different news sources within each major news story
        link=None
        try:
            link = re.search('(?P<url>https?:\/\/[^\s]+)', str(substory_list)).group('url')
        except : continue
        link = link[:-1]
        d['Source_Link'][i]=link
    extract_articles(d['Source_Headline'])

```

Sau khi có bảng data ta chạy hàm `extract_articles` với đầu vào là các link từ cột `source_headline` trong bảng data , trong đó ta gọi hàm `soup_basics` cho các link , với kết quả trả về ta truy cập lớp class `read-more-story` sau đó lấy link bài viết gốc từ trong thẻ đó ( vì các trang web có thể khác nhau nên ta sử dụng cặp lệnh `try except` để khi gặp những trường hợp trang web không phản hồi thì sẽ được bỏ qua và tiếp tục vòng lặp )

### From The Center

U.S. Attorney General William Barr has authorized the Department of Justice to investigate "substantial allegations" of voting irregularities in this year's presidential election despite little evidence of widespread voter fraud, according to a memo obtained by the Associated Press.

In the memo to U.S. attorneys, Barr said investigations "may be conducted if there are clear and apparently-credible allegations of irregularities that, if true, could potentially impact the outcome of a federal election in an individual State."

Read Full Story

Link bài viết gốc nằm trong thẻ `a href` thuộc class `read-more-story`

```
<div class="read-more-story">
  <a href="https://www.newsweek.com/barr-tells-federal-prosecutors-investigate-clear-instances-voting-irregularities-sidestepping-1546180" target="_blank" rel="noopener">Read full story</a>
</div>
```

## 3.4. CRAWL IMAGE

```
# get all image from article links
d['Image_Link']=0

def extract_articles(link):
    for i in range(len(d['Source_Link'])):
        try:
            soup = soup_basics(d['Source_Link'][i])
            link=None
            # locate relevant information within the extracted page
            story_image = soup.find(property="og:image")

            # loop through the different news sources within each major news story

            link = re.search('(P<url>https?://[^s]+)', str(story_image)).group('url')
        except: continue
        link = link[:-1]
        d['Image_Link'][i]=link
    extract_articles(d['Source_Link'])
```

Sau khi cập nhật là cột `source_link` chứa link gốc của bài viết, ta chạy hàm `extract_articles` với đầu vào là các link từ cột `source_link` trong bảng data , trong đó ta gọi hàm `soup_basics` cho các link , với kết quả trả về ta truy cập lớp `property og:image` sau đó

lấy link như từ bài viết gốc từ trong lớp đó ( tương tự như lấy link bài viết gốc , một số trang web có thể chặn việc khai thác ảnh hoặc các trang web khác nhau định dạng cũng sẽ khác nhau nên ta sẽ đặt trong cặp lệnh try except )

Cuối cùng ta có được bảng data:

|     | Date                      | Description                                       | Source_Name                                       | Source_Bias | Source_HeadLine                                   | Source_Link                                       | Image_Link  |
|-----|---------------------------|---|---|-------------|---|---|---|
| 0   | 2018-03-08T19:34:58-08:00 | The people who like the sort of tariffs that p... | OPINION: Trump's risky call for protests          | Left        | https://www.allsides.com/news/2018-03-08-1534/... | http://money.cnn.com/2018/03/08/news/economy/...  | https://i2.cdn.turner.com/money/dam/assets/180... |
| 1   | 2020-05-28T07:02:58-07:00 | President Donald Trump is preparing to sign an... | OPINION: Trump's risky call for protests          | Left        | https://www.allsides.com/news/2020-05-28-0702/... | https://agnews.com/630f9ebd0f5d3d3a70e7374f...    | https://storage.googleapis.com/af-prod-media/...  |
| 2   | 2019-12-12T06:52:09-08:00 | The Justice Department's top watchdog on Wedne... | OPINION: Donald Trump has committed a lot of s... | Lean Left   | https://www.allsides.com/news/2019-12-12-0659/... | https://www.politico.com/news/2019/12/11/horow... | https://cf-images-us-east-1-prod-boltdns.net/...  |
| 3   | 2018-07-12T12:49:56-07:00 | President Trump kept everyone guessing to the ... | Trump Indictment Could Be a 2024 Cash Cow         | Center      | https://www.allsides.com/news/2018-07-12-1249/... | https://www.wsj.com/articles/kavanaugh-for-the... | https://s.wsj.net/img/meta/wsj-social-share.png   |
| 4   | 2019-10-17T06:20:50-07:00 | As President Donald Trump continues to fill hi... | Police surround NY Courthouse and DC Capitol i... | Center      | https://www.allsides.com/news/2019-10-17-0620/... | https://abcnews.go.com/Politics/exclusive-hid...  | https://s.abcnews.com/images/US/Eiden-intervie... |
| 419 | 2018-03-08T14:19:59-08:00 | President Donald Trump on Thursday signed proc... | As Supreme Court takes up Trump plan to end DA... | Lean Left   | https://www.allsides.com/news/2018-03-08-1419/... | https://www.wsj.com/articles/trump-to-meet-mat... | https://s.wsj.net/img/meta/wsj-social-share.png   |
| 420 | 2018-04-19T15:20:47-07:00 | Donald Trump was warned. Seven days before his... | FACT CHECK: The Data on DACA and Crime            | Center      | https://www.allsides.com/news/2018-04-19-1519/... | https://www.wsj.com/articles/a-higher-sandtim...  | https://s.wsj.net/img/meta/wsj-social-share.png   |
| 421 | 2020-01-30T09:40:20-08:00 | President Trump's legal team offered a startli... | 3 ways the Supreme Court could decide DACA's fate | Left        | https://www.allsides.com/news/2020-01-30-0940/... | https://www.washingtonpost.com/politics/trump...  | https://www.washingtonpost.com/wp-apps/imrs.ph... |
| 422 | 2022-08-11T07:42:09-07:00 | As the late political philosopher Hans Gruber ... | Don't count on the Senate to save Dreamers fro... | Lean Left   | https://www.allsides.com/news/2022-08-11-0742/... | https://www.nationalreview.com/2022/08/fbi-has... | https://www.nationalreview.com/wp-content/uplo... |
| 423 | 2019-05-12T08:04:24-07:00 | The Republican party was struggling to heal s...  | Supreme Court Justices couldn't stop interrup...  | Center      | https://www.allsides.com/news/2019-05-12-1224/... | http://www.theguardian.com/us-news/2019/may/12... | https://i.guim.co.uk/img/media/7d6303c1e6838ec... |

Ta xuất tất cả các link từ cột source link thành file 'link\_file.txt' để crawl text.

Xoá bỏ những hàng có chứa missing value và xuất thành file ' allsides-content-f.csv'.

```

file_1 = list(d.Source_Link)

with open('link_file.txt', 'w') as file:
    # Iterate over the list and write each string to a new line in the file
    for string in file_1:
        file.write(string + '\n')

d = d.dropna()

d.to_csv("allsides-content-f.csv", index=False)

```

### 3.5. CRAWL TEXT

```

!pip install news-please

Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.9/dist-packages (from nltk>=3.2.1->newspaper3k>=0.2.8->news-please) (2022.10.31)
Requirement already satisfied: joblib in /usr/local/lib/python3.9/dist-packages (from nltk>=3.2.1->newspaper3k>=0.2.8->news-please) (1.1.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.9/dist-packages (from requests>=2.10.0->newspaper3k>=0.2.8->news-please) (3.4)
Requirement already satisfied: charset-normalizer<=2.0.0 in /usr/local/lib/python3.9/dist-packages (from requests>=2.10.0->newspaper3k>=0.2.8->news-please) (2.0.12)
Requirement already satisfied: pynini>=0.1.3 in /usr/local/lib/python3.9/dist-packages (from rsac4.8,>=3.1.2->newscli>=1.11.117->news-please) (0.4.8)
Requirement already satisfied: pyasn1-modules in /usr/local/lib/python3.9/dist-packages (from service-identity>=18.1.0->Scrapy>=1.1.0->news-please) (0.2.8)
Requirement already satisfied: attrs>=19.1.0 in /usr/local/lib/python3.9/dist-packages (from service-identity>=18.1.0->Scrapy>=1.1.0->news-please) (22.2.0)
Requirement already satisfied: filelock>=3.0.8 in /usr/local/lib/python3.9/dist-packages (from tlextract>=2.0.1->newspaper3k>=0.2.8->news-please) (3.10.7)
Collecting requests-file>=1.4
  Downloading requests_file-1.5.1-py2.py3-none-any.whl (3.7 kB)
Collecting Automat>=0.8.0
  Downloading Automat-22.10.0-py2.py3-none-any.whl (26 kB)
Requirement already satisfied: typing-extensions>=3.6.5 in /usr/local/lib/python3.9/dist-packages (from Twisted>=18.9.0->Scrapy>=1.1.0->news-please) (4.5.0)
Collecting constantly>=15.1
  Downloading constantly-15.1.0-py2.py3-none-any.whl (7.9 kB)
Collecting incremental>=21.3.0
  Downloading incremental-22.10.0-py2.py3-none-any.whl (16 kB)
Collecting hyperlink>=17.1.1
  Downloading hyperlink-21.0.0-py2.py3-none-any.whl (74 kB)

```

- Để có thể crawl full page text từ link url có sẵn, ta sẽ cần sử dụng thư viện newsplease nên ta sẽ install và import nó.

| Date                      | Description                                       | Source_Name                                       | Source_Bias | Source_Headline  | Source_Link   |
|---------------------------|---|---|-------------|--|---|
| 2018-03-08T15:34:58-08:00 | The people who like the sort of tariffs that P... | OPINION: Trump's risky call for protests          | Left        | <a href="https://www.allsides.com/news/2018-03-08-1534/">https://www.allsides.com/news/2018-03-08-1534/...</a> | <a href="http://money.cnn.com/2018/03/05/news/economy/t...">http://money.cnn.com/2018/03/05/news/economy/t...</a> <a href="https://i2.cdn.turner.com/money/dam/a">https://i2.cdn.turner.com/money/dam/a</a> |
| 2020-05-28T07:02:58-07:00 | President Donald Trump is preparing to sign an... | OPINION: Trump's risky call for protests          | Left        | <a href="https://www.allsides.com/news/2020-05-28-0702/">https://www.allsides.com/news/2020-05-28-0702/...</a> | <a href="https://apnews.com/fc30f9ebdf9d3d33a870e7374f...">https://apnews.com/fc30f9ebdf9d3d33a870e7374f...</a> <a href="https://storage.googleapis.com/afs-pr">https://storage.googleapis.com/afs-pr</a>   |
| 2019-12-12T06:52:09-08:00 | The Justice Department's top watchdog on Wedne... | OPINION: Donald Trump has committed a lot of s... | Lean Left   | <a href="https://www.allsides.com/news/2019-12-12-0651/">https://www.allsides.com/news/2019-12-12-0651/...</a> | <a href="https://www.politico.com/news/2019/12/11/horow...">https://www.politico.com/news/2019/12/11/horow...</a> <a href="https://cf-images.us-east-1.prod.bol">https://cf-images.us-east-1.prod.bol</a>   |

- Tổng quan về data mà ta sẽ dùng, trong data này thì ta chỉ sử dụng trường Source\_Link để tiến hành.

```
import math
def chuck(xs, n):
    assert n > 0
    L = len(xs)
    s, r = divmod(L, n)
    widths = chain(repeat(s+1, r), repeat(s, n-r))
    offsets = accumulate(chain((0,), widths))
    b, e = tee(offsets)
    next(e)
    return [xs[s] for s in map(slice, b, e)]

batch = chuck(news_links, 270)

batch[3:7]
```

- Ta sẽ tạo một hàm chuck để có thể chia các link url thành các list nhỏ để khi ta gọi method from\_urls() của newplease thì ta có thể xử lý và lấy các thông tin cần thiết một cách dễ dàng hơn.

```
def article_crawler():
    # crawler
    n = 0
    for i in range(0, len(batch)):
        try:
            slice = batch[i]
            # print slice
            slice_name = str(i) + '-NewsPlease-articleCrawl.p'
            article_information = NewsPlease.from_urls(slice)
            pickle.dump(article_information, open(slice_name, 'wb'))
            n += 1
        except:
            continue

article_crawler()
```

- Sau khi đã cho các url thành các list nhỏ trong 1 list lớn thì ta tiến hành crawl các thông tin cần thiết qua hàm `article_crawler` và chuyển đổi nó thành 1 chuỗi các byte để lưu vào từng file có định dạng '-NewsPlease-articleCrawl.p'.

```
def make_unique(url_list):
    # Not order preserving
    unique = set(url_list)
    return list(unique)

def check_data(filepath):
    scraped = []
    not_scraped = []

    for i in range(0, (math.ceil(len(news_links)/2))):
        try:
            #file_path = filepath+"/crawl/"
            open_crawl = pickle.load(open(str(i) + "-NewsPlease-articleCrawl.p", "rb"))
            for url in open_crawl:
                text = open_crawl[str(url)].maintext
                if text == None:
                    not_scraped.append(url)
                else:
                    scraped.append(url)
            except FileNotFoundError:
                continue

    scraped = make_unique(scraped)
    return scraped, not_scraped
```

```
success, fail = check_data(filepath)

# analyze data collection success
def percentage(part, whole):
    percent = 100 * float(part)/float(whole)
    format = "{0:.2f}".format(percent)
    return format+'%'

print("The extraction process yielded "
      + str(len(success)) + " articles, or "
      + percentage(len(success),len(news_links))
      + " of the total.")
```

The extraction process yielded 321 articles, or 75.71% of the total.

- Sau khi đã cào được các thông tin cần thiết và lưu vào file rồi thì ta sẽ tiến hành xem và kiểm tra xem có bao nhiêu link có thể cào được thông tin. Và ta thấy là có 75.71% tức 321 bài viết là cào được các thông tin từ link đầy đủ còn 103 link còn lại không thể cào được thông tin. Nên nhóm em cho rằng có thể là do link bị thay đổi hoặc link bài viết đã bị chết.

```
def get_data():
    news_dict = {}

    remove_list = ['www.', 'www1.', '.com', '.gov', '.org', 'beta.', '.eu',
                    '.co.uk', 'europe', 'gma', 'blogs', 'in.', 'm.',
                    'eclipse2017.', 'money', 'insider', 'news.', 'finance.',
                    'www1.']

    for i in range(0, 220):
        try:
            #file_path = filepath + "/crawl/"
            open_crawl = pickle.load(open(str(i) + "-NewsPlease-articleCrawl.p", "rb"))

            for url in open_crawl:
                text = open_crawl[str(url)].maintext
                if text != None:
                    title = open_crawl[str(url)].title
                    authors = ', '.join(open_crawl[str(url)].authors)
                    source = open_crawl[str(url)].source_domain

                    for seq in remove_list:
                        if seq in source:
                            source = source.replace(seq, "")

                    date = open_crawl[str(url)].date_publish

                    news_dict[str(url)] = [source, title, authors, date, text]
```

```
except FileNotFoundError:
    continue

return news_dict

all_news = get_data()
print(all_news)

('https://agencies.com/fc30f9ebdfff9d33a870e737d48ffaf': ['apcom', 'Trump escalates war on Twitter, social media protections', 'Zeke Miller', datetime.datetime(2021, 4, 20, 12, 45, 5
```

- Cuối cùng, ta sẽ chuyển các file chứa chuỗi byte các dữ liệu thành các dữ liệu hoàn chỉnh và lưu nó vào trong một dictionary có keys là link url của bài viết đầy đủ và values là các thông tin như tên miền, tiêu đề, tác giả, ngày công bố, nội dung.

```
data['main_headline'] = data['Source_Link'].apply(lambda x: all_news[x][1] if x in all_news else None)
data['text'] = data['Source_Link'].apply(lambda x: all_news[x][4] if x in all_news else None)
data['author'] = data['Source_Link'].apply(lambda x: all_news[x][2] if x in all_news else None)
```

- Ta chỉ cần lấy các giá trị cần thiết của từng keys trong dictionary và so sánh với link url của Source\_Link để đưa các giá trị đó vào dataframe ban đầu (data) và tạo thành 3 cột mới là main\_headline, text và author. Đối với những link url của bài viết không đầy đủ sẽ không cào được thông tin gì, nên ta sẽ thay các giá trị của 3 cột mới main\_headline, text và author đối với những link url đó là None.

```
data['Source_Headline'] = data['Source_Link'].apply(extract_domain)
data = data.rename(columns={'Date': 'date', 'Description': 'description', 'Source_Name': 'source name', 'Source_Bias': 'bias',
                             'Source_Headline': 'source', 'Source_Link': 'link'})
data.head()
```

```

data = data.drop(data[data.text == 'None'].index)
data = data.drop(data[data.text == 'Mixed'].index)

data['text_len'] = data.apply(lambda row: len(row.text), axis=1)

data = data.drop(data[data.text_len > 20000].index)
data = data.drop(data[data.text_len < 250].index)

# write to csv
data.to_csv('news-corpus-df.csv', index=False)
data.head()

```

- Tiếp theo, ta thay đổi tên các cột theo mong muốn và đếm độ dài của từng article và lưu nó thành 1 cột mới trong dataframe data và lưu nó vào file csv. Dưới đây là kết quả cuối cùng sau khi hoàn tất công việc cào dữ liệu.

| data                      | description                                       | source name                                       | bias      | source   | link  | image_link  | main_headline                                     | text   | author        | text_len |
|---------------------------|---|---|-----------|----------|---|---|---|--|---------------|----------|
| 2018-03-08T15:34:58-08:00 | The people who like the sort of tariffs that P... | OPINION: Trump's risky call for protests          | Left      | cnn      | http://money.cnn.com/2018/03/05/news/economy/t... | https://i2.cdn.turner.com/money/dam/assets/180... | Who likes tariffs? Generally speaking, it's no... | The people who like the sort of tariffs that P...  | Chris Isidore | 3013     |
| 2020-05-28T07:02:58-07:00 | President Donald Trump is preparing to sign an... | OPINION: Trump's risky call for protests          | Left      | apnews   | https://apnews.com/c30f9ebdf9d3d33a670e7374f...   | https://storage.googleapis.com/afs-prod/media/... | Trump escalates war on Twitter, social media p... | FILE - In this Oct. 11, 2017, file photo, MSNBC... | Zake Miller   | 8726     |
| 2019-12-12T06:52:09-08:00 | The Justice Department's top watchdog on Wedne... | OPINION: Donald Trump has committed a lot of s... | Lean Left | politico | https://www.politico.com/news/2019/12/11/horow... | https://cf-images-us-east-1-prod-boltdns.net/v... | Horowitz pushes back at Barr over basis for T...  | Graham was also critical of the FBI's decision...  |               | 6234     |



## CHƯƠNG 4. XỬ LÝ VĂN BẢN , PHÂN TÍCH MỨC ĐỘ CẢM XÚC & ỨNG DỤNG CONTENT-BASED IMAGE SEARCH

### 4.1. LÀM SẠCH VĂN BẢN

- Định nghĩa hàm `read_data` để đọc dữ liệu đầu vào, các dữ liệu bao gồm có 'main\_headline', 'bias', 'text', 'text\_len', 'source', 'Image\_link'
- Lấy các giá trị chữ có trong cột 'bias' bỏ đi các khoảng trắng (hàm `strip()`). Gán nhãn các 'bias' theo số như sau:
  - 1: Left
  - 2: Lean Left
  - 3: Center
  - 4: Lean Right
  - 5: Right
- Bỏ các dòng có giá Null/NaN và các dòng có 'text\_len' (độ dài bài báo) nhỏ hơn 500 từ.

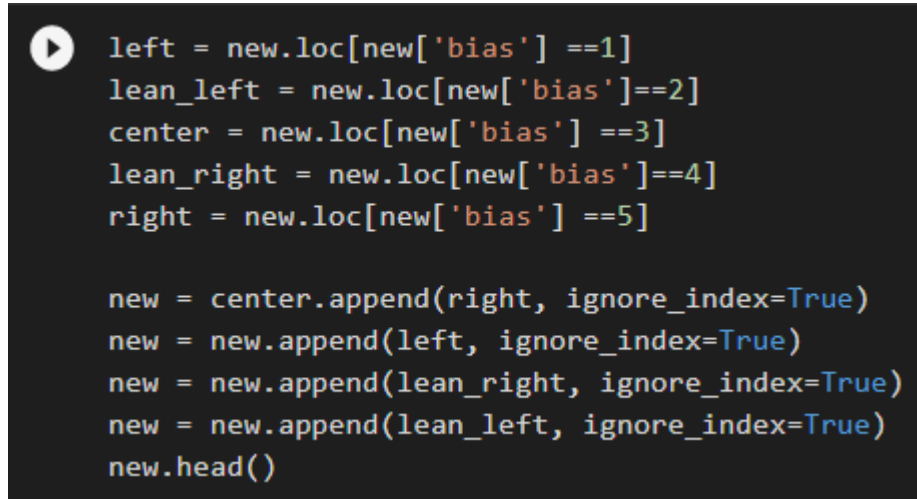
```
def read_data(filename):  
    # read in csv  
    df = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/(SB 142023)news-corpus-df.csv')  
  
    # drop text under 500 words  
    df = df.drop(df[df.text_len < 500].index)  
    df.dropna(inplace = True)  
  
    # limit df content to bias, text, headline, and source  
    df = df.loc[:, ['main_headline', 'bias', 'text', 'text_len', 'source', 'Image_link']]  
  
    # cleaning the 'bias' column of unnecessary white space  
    df['bias'] = df['bias'].apply(lambda x: x.strip())  
  
    # convert bias label to number  
    df['bias'] = df['bias'].replace({'Left': 1, 'Center': 3, 'Right': 5})  
    df['bias'] = df['bias'].replace({'Lean Left': 2, 'Lean Right': 4})  
  
    return df  
  
# read in file and preview  
new = read_data('/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/(SB 142023)news-corpus-df.csv')  
new.head()
```

Hình 4.1.1. hàm `read_data`

|   | main_headline                                     | bias |   | text  | text_len | source  | Image_link |
|---|---|------|---|-------|----------|---|------------|
| 0 | Who likes tariffs? Generally speaking, it's no... | 1    | The people who like the sort of tariffs that P... | 3013  | cnn      | https://2.cdn.turner.com/money/dam/assets/180...  |            |
| 1 | Trump escalates war on Twitter, social media p... | 1    | FILE - In this Oct. 11, 2017, file photo, MSNB... | 8726  | apnews   | https://storage.googleapis.com/afs-prod/media/... |            |
| 3 | Kavanaugh for the Court                           | 3    | Judge Brett Kavanaugh speaks after being nomin... | 1062  | wsj      | https://s.wsj.net/img/meta/wsj-social-share.png   |            |
| 4 | Exclusive: 'Ym here' Hunter Biden hits back ...   | 3    | As President Donald Trump continues to fill hi... | 15428 | go       | https://s.abcnews.com/images/US/Biden-intervie... |            |
| 5 | Trump made Twitter his megaphone. A fact check... | 5    | When President Donald Trump felt the need to L... | 5703  | nbcnews  | https://media-cldnry.s-nbcnews.com/image/uploa... |            |

Hình 4.1.2. bộ dữ liệu

- Sắp xếp các dòng lại theo cụm bias, với thứ tự là 3 (Center), 5 (Right), 1 (Left), 4 (Lean Right), 2 (Lean Left)



```

left = new.loc[new['bias'] ==1]
lean_left = new.loc[new['bias']==2]
center = new.loc[new['bias'] ==3]
lean_right = new.loc[new['bias']==4]
right = new.loc[new['bias'] ==5]

new = center.append(right, ignore_index=True)
new = new.append(left, ignore_index=True)
new = new.append(lean_right, ignore_index=True)
new = new.append(lean_left, ignore_index=True)
new.head()

```

Hình 4.1.3. Sắp xếp các dòng theo cụm bias

- Tạo hàm text\_prepare() xử lý stopwords:
  - text.lower(): chuyển tất cả các chữ về dạng chữ thường
  - text.replace('\n', ' '): chuyển các ký tự xuống dòng '\n' thành khoảng trắng
  - letters = list(string.ascii\_lowercase): tạo ra list chứa các từ ở dạng chữ thường trong bảng chữ cái ASCII
  - Tạo ra list numbers chứa các số (dạng đơn vị) từ 0-9
  - Tạo ra list banned chứa các ký tự cấm (ký tự đặc biệt), sau đó nối các từ trong list banned với nhau với dấu chấm câu, kèm theo số từ list
  - Tạo list boilerplate chứa các từ thường xuất hiện trên các trang báo điện tử nhưng không có ý nghĩa trong các bài báo
  - Tạo set stoplist chứa tập hợp các stopwords trong đó bao gồm các stopwords tiếng Anh stopwords.words('english'), các từ trong list boilerplate và các từ trong list letters
  - Tạo dictionary translation\_table ánh xạ các ký tự/từ trong list banned lên khoảng trắng. Sau đó dùng dictionary này để thay thế các từ thuộc list banned trong text thành khoảng trắng.
  - Thay toàn bộ các khoảng trắng liên tiếp thành 1 khoảng trắng duy nhất
  - Chia văn bản thành các từ riêng lẻ, xóa tất cả các stopwords thuộc stoplist và nối các từ còn lại với nhau bằng dấu cách.
  - Trả về văn bản đã được xử lý

```
def text_prepare(text):
    """
    text: a string
    return: modified initial string
    """
    text = text.lower()
    text = text.replace("\n", ' ')

    letters = list(string.ascii_lowercase)
    numbers = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']
    banned = ["'", '"', "-", "_", ":", ";", ",", ".", "!", "[", "]",
              '(', ')', '{', '}', '\\', '[', ']', '|', '@', ' ', ' ', '+', '-']
    banned = ''.join(banned) + string.punctuation + ''.join(numbers)
    boilerplate = [' ', 'https', 'http', 'www', 's', '-', '/', 'playback', 'get', 'mr', 'mrs', 'ms', 'dr', 'prof', 'news', 'report', 'unsub',
                   'contributed', 'advertisement', 'the washington', '8', 'follow', 'copyright', 'mrs', 'photo', 'to', 'also', 'times', 'for']
    stop_list = set(stopwords.words('english') + boilerplate + letters)

    translation_table = dict.fromkeys(map(ord, banned), ' ')
    text = text.translate(translation_table)
    text = re.sub(' +', ' ', text)
    text = ' '.join([word for word in text.split() if word not in stop_list])
    return text
```

Hình 4.1.4. Hàm xử lý stopwords

```
# rewrite df with cleaned text
for i in range(0, len(new)):
    new.at[i, 'text'] = text_prepare(new.at[i, 'text'])
    new.at[i, 'main_headline'] = text_prepare(new.at[i, 'main_headline'])
new.head()
```

Hình 4.1.5. Trả về cột ‘text’ và ‘main headline’ đã được xử lý stopwords

|   | main_headline                                     | bias | text  | text_len | source   | Image_link  |
|---|---|------|---|----------|----------|---|
| 0 | kavanaugh court                                   | 3    | judge brett kavanaugh speaks nominated preside... | 1062     | wsj      | <a href="https://s.wsj.net/img/meta/wsj-social-share.png">https://s.wsj.net/img/meta/wsj-social-share.png</a>       |
| 1 | exclusive hunter Biden hits back trump taunt e... | 3    | president donald trump continues fall twitter ... | 15428    | go       | <a href="https://s.abnnews.com/images/US/Biden-intervie...">https://s.abnnews.com/images/US/Biden-intervie...</a>   |
| 2 | republican reps cite anniversary criticizing t... | 3    | washington cnn two republican members congress... | 3281     | cnn      | <a href="https://media.cnn.com/api/v1/images/stellar/pr...">https://media.cnn.com/api/v1/images/stellar/pr...</a>   |
| 3 | brazilian president demands apology accepting ... | 3    | brazilian president jair bolsonaro tuesday acc... | 4322     | nbcbnews | <a href="https://media-cldnry.s-nbcnews.com/image/upload...">https://media-cldnry.s-nbcnews.com/image/upload...</a> |
| 4 | amazon fires exactly burning earth lungs exper... | 3    | earth lungs fire burning version politicians j... | 2208     | foxnews  | <a href="https://static.foxnews.com/foxnews.com/content...">https://static.foxnews.com/foxnews.com/content...</a>   |

Hình 4.1.6. bộ dữ liệu sau khi xử lý stopwords

- Tạo hàm `stem_word()` để thực hiện stemming (loại bỏ các hậu tố của các từ như -s/-es/-ing có trong 'text')

```
stemmer = PorterStemmer()
lemmatizer = WordNetLemmatizer()

def stem_word(word):
    """
    Stem a word using the Porter stemming algorithm
    """
    return stemmer.stem(word)
```

Hình 4.1.7. Hàm stem\_word()

- Tạo hàm `lemmatize_word()` để thực hiện lemmatization (phương pháp trả về từ gốc của một từ, từ gốc này có trong từ điển), hàm này sử dụng `WordNetLemmatizer` từ thư viện `nltk.stem`. Đầu tiên sử dụng `synsets` để tìm ra

danh sách các từ đồng nghĩa với từ đang được xử lý. Nếu không có sẽ mặc định từ đó là một danh từ.

- Sau đó, POS của từ được xác định bằng cách lấy phần pos tag của tập hợp đầu tiên trong danh sách (nếu nó tồn tại). Pos tag được thể hiện bằng một trong các ký tự sau: 'J' cho tính từ, 'V' cho động từ, 'N' cho danh từ và 'R' cho trạng từ. Nếu một phần của text không phải là một trong những điều này, hàm sẽ cho từ đó là một danh từ.
- Tiếp theo, một phần của pos tag được ánh xạ tới hằng số WordNet thích hợp. Cuối cùng, phương thức lemmatize() của lemmatizer được gọi với từ đang được xử lý và một phần của pos tag của nó làm đối số để thực hiện việc từ vựng hóa thực sự. Bộ đề kết quả (dạng cơ bản của từ) được hàm trả về.

```
def lemmatize_word(word):  
    """  
    Lemmatize a word using the WordNet lemmatization algorithm  
    """  
    # Get the part of speech for the word  
    pos = wordnet.synsets(word)[0].pos() if wordnet.synsets(word) else 'n'  
    # Map the part of speech to the corresponding WordNet constant  
    if pos.startswith('J'):  
        pos = wordnet.ADJ  
    elif pos.startswith('V'):  
        pos = wordnet.VERB  
    elif pos.startswith('N'):  
        pos = wordnet.NOUN  
    elif pos.startswith('R'):  
        pos = wordnet.ADV  
    else:  
        pos = wordnet.NOUN  
    # Lemmatize the word using the appropriate part of speech  
    return lemmatizer.lemmatize(word, pos)
```

Hình 4.1.8. Hàm lemmatize\_word()

- Tạo hàm preprocess\_text() để kết hợp hàm word\_tokenize() có sẵn trong thư viện nltk (token hóa các từ), stem\_word() và lemmatize\_word() để xử lý từng từ có trong 1 text sau đó ghép các từ lại như 'text' ban đầu:

```
def preprocess_text(text):  
    """  
    Preprocess a block of text by tokenizing, stemming, and lemmatizing each word  
    """  
    tokens = word_tokenize(text)  
    stemmed = [stem_word(token) for token in tokens]  
    lemmatized = [lemmatize_word(token) for token in stemmed]  
    return ' '.join(lemmatized)
```

Hình 4.1.9. Hàm preprocess\_text()

```
for i in range(0, len(new)):
    new.at[i, 'text'] = preprocess_text(new.at[i, 'text'])
    new.at[i, 'main_headline'] = preprocess_text(new.at[i, 'main_headline'])

new.head()
```

Hình 4.1.10. Thực hiện hàm preprocess\_text lên từng dòng của bộ data

|   | main_headline                                     | bias |   | text  | text_len | source  | Image_link |
|---|---|------|---|-------|----------|---|------------|
| 0 | kavanaugh court                                   | 3    | judg brett kavanaugh speak nomin presid donald... | 1062  | wsj      | <a href="https://s.wsj.net/img/meta/wsj-social-share.png">https://s.wsj.net/img/meta/wsj-social-share.png</a>     |            |
| 1 | exclus hunter Biden hit back trump taunt exclu... | 3    | presid donald trump continu fill twitter feed ... | 15428 | go       | <a href="https://s.abcnews.com/images/US/Biden-intervie...">https://s.abcnews.com/images/US/Biden-intervie...</a> |            |
| 2 | republican rep cite anniversari critic trump d... | 3    | washington cnn two republican member congress ... | 3281  | cnn      | <a href="https://media.cnn.com/api/v1/images/stellar/pr...">https://media.cnn.com/api/v1/images/stellar/pr...</a> |            |
| 3 | brazilian presid demand apolog accept help fig... | 3    | brazilian presid jair bolsonaro tuesday accept... | 4322  | nbcnews  | <a href="https://media-cldnry.s-nbcnews.com/image/uploa...">https://media-cldnry.s-nbcnews.com/image/uploa...</a> |            |
| 4 | amazon fire exactli burn earth lung expert say    | 3    | earth lung fire burn version politician journa... | 2208  | foxnews  | <a href="https://static.foxnews.com/foxnews.com/content...">https://static.foxnews.com/foxnews.com/content...</a> |            |

Hình 4.1.11. Bộ data sau khi làm sạch văn bản 'text'

## 4.2. ĐÁNH GIÁ CẢM XÚC VĂN BẢN (SENTIMENT ANALYSIS)

- Tạo vòng lặp duyệt qua từng dòng của dataframe new['text']
  - Điểm sentiment\_score() (điểm đánh giá cảm xúc dựa trên điểm số định sẵn cho từng từ trong thư viện AFINN, được tính bằng cách cộng điểm số từng từ lại để cho ra tổng điểm của một đoạn văn bản, có thể âm dương tùy vào các từ có trong đoạn văn bản) và magnitude\_score() (độ lớn của sentiment\_score(), sử dụng abs(sentiment\_score()), thể hiện mức độ, độ lớn của cảm xúc được thể hiện trong đoạn văn bản, magnitude\_score() càng lớn thì độ lớn cảm xúc được thể hiện cũng càng lớn)
  - Mỗi vòng lặp xử lý điểm số cho từng dòng trong dataframe new có 1 khoảng trễ rơi vào 0.1s. Độ trễ này được thêm vào để tránh quá tải API AFINN với quá nhiều yêu cầu cùng một lúc. Bằng cách thêm độ trễ này, vòng lặp sẽ tạm dừng trong một khoảng thời gian ngắn giữa mỗi lần lặp, điều này giúp trải đều các yêu cầu theo thời gian và ngăn API bị quá tải với quá nhiều yêu cầu cùng một lúc. Được thực hiện nhờ vào dòng lệnh time.sleep(.100)

```

from afinn import Afinn
import re
afinn = Afinn()
sentiment_list = []
magnitude_list = []

for text in new['text']:
    # Compute sentiment score
    sentiment_score = afinn.score(text)
    sentiment_list.append(sentiment_score)
    # Compute magnitude score
    magnitude_score = sum(abs(afinn.score(word)) for word in text.split() if word in afinn._dict)
    magnitude_list.append(magnitude_score)

    # wait a second to add delay to query
    time.sleep(.100)

# Add sentiment information to data frame
new = new.assign(sentiment=sentiment_list)
new = new.assign(magnitude=magnitude_list)
new.head(11)

```

Hình 4.2.1. Đánh giá sentiment\_score() và magnitude\_score()

- Đưa các điểm số đã tính cho từng dòng tạo thành 2 cột mới là “sentiment” và “magnitude” vào trong dataframe new.

|    | main_headline                                      | bias | text  | text_len | source         | image_link  | sentiment | magnitude |      |
|----|--|------|---|----------|----------------|---|-----------|-----------|------|
| 0  | kavanaugh court                                    | 3    | judg brett kavanaugh speak nomin presid donald... | 1062     | wsj            | https://s.wsj.net/img/meta/wsj-social-share.png   | 4.0       | 6.0       |      |
| 1  | exclus hunter biden hit back trump taunt exclu...  | 3    | presid donald trump continu fill twitter feed ... | 15428    | go             | https://s.abcnews.com/images/US/Biden-intervie... | 59.0      | 171.0     |      |
| 2  | republican rep cite anniversari critic trump d...  | 3    | washington cnn two republican member congress ... | 3281     | cnn            | https://media.cnn.com/api/v1/images/stellar/pr... | -14.0     | 40.0      |      |
| 3  | brazilian presid demand apolog accept help fig ... | 3    | brazilian presid jair bolsonaro tuesday accept... | 4322     | nbcnews        | https://media-cldnry.s-nbcnews.com/image/uploa... | 8.0       | 82.0      |      |
| 4  | amazon fire exactli burn earth lung expert say     | 3    | earth lung fire burn version politician journa... | 2208     | foxnews        | https://static.foxnews.com/foxnews.com/content... | -16.0     | 28.0      |      |
| 5  | racist tweet medium grappl label trump latest ...  | 3    | comment stori comment gift articl time call st... | 6298     | washingtonpost | https://www.washingtonpost.com/wp-apps/imrs.ph... | -76.0     | 118.0     |      |
| 6  | republican longer fear tucker carlson              | 3    | widespread republican critic tucker carlson re... | 4330     | newsweek       |   | 0         | -63.0     | 73.0 |
| 7  | reopen hillari email case matter                   | 3    | make mistak enjoy reviv fbi interest hillari c... | 6279     | townhall       | https://media.townhall.com/cdn/hodl/2016/272/7... | 23.0      | 97.0      |      |
| 8  | biden say democraci prevail elector colleg for...  | 3    | presid elect joe biden emphas uniti speech mon... | 1543     | axios          | https://images.axios.com/0PwpU3ma9LrM-cONebxni... | 16.0      | 24.0      |      |
| 9  | twitter fact check trump trump cri free speech     | 3    | twitter trump collis cours truth twitter appli... | 5933     | usatoday       | https://www.gannett-cdn.com/presto/2019/07/26/... | -46.0     | 96.0      |      |
| 10 | opinion trump get right afghanistan pakistan       | 3    | gift articl presid trump fort myer speech poss... | 6944     | washingtonpost | https://www.washingtonpost.com/wp-apps/imrs.ph... | -26.0     | 102.0     |      |

Hình 4.2.2. Dataframe new sau khi tính điểm số sentiment và magnitude

- Gán các điểm “sentiment” dương là “positive”, âm là “negative” và 0 là “neutral”. Tạo thành một cột mới là “senti” chứa các dữ liệu đã được gán:

```
def score_to_sentiment(score):
    if score > 0:
        return 'positive'
    elif score < 0:
        return 'negative'
    else:
        return 'neutral'

new['senti'] = new['sentiment'].apply(score_to_sentiment)
new.head()
```

Hình 4.2.3. Gán các điểm số thành đánh giá cảm xúc

|   | main_headline  | bias | text   | text_len | source  | Image_link   | sentiment | magnitude | senti    |
|---|--|------|--|----------|---------|--|-----------|-----------|----------|
| 0 | kavanaugh court                                      | 3    | judg brett kavanaugh speak nomin<br>presid donald... | 1062     | wsj     | https://s.wsj.net/img/meta/wsj-social-share.png        | 4.0       | 6.0       | positive |
| 1 | exclus hunter biden hit back trump<br>taunt exclu... | 3    | presid donald trump continu fill<br>twitter feed ... | 15428    | go      | https://s.abcnews.com/images/US/Biden-<br>interview... | 59.0      | 171.0     | positive |
| 2 | republican rep cite anniversari<br>critic trump d... | 3    | washington cnn two republican<br>member congress ... | 3281     | cnn     | https://media.cnn.com/api/v1/images/stellar/pr...      | -14.0     | 40.0      | negative |
| 3 | brazilian presid demand apolog<br>accept help fig... | 3    | brazilian presid jair bolsonaro<br>tuesday accept... | 4322     | nbcnews | https://media-cdnny.s-<br>nbcnews.com/image/uploa...   | 8.0       | 82.0      | positive |
| 4 | amazon fire exactli burn earth lung<br>expert say    | 3    | earth lung fire burn version<br>politician journa... | 2208     | foxnews | https://static.foxnews.com/foxnews.com/content...      | -16.0     | 28.0      | negative |

Hình 4.2.4. Bộ dữ liệu sau khi thêm cột đánh giá cảm xúc

### Nhận xét đánh giá, vẽ biểu đồ:

```
new['senti'].value_counts()

negative    164
positive     88
neutral     13
Name: senti, dtype: int64
```

Hình 4.2.5. Thống kê số cảm xúc tích cực, tiêu cực và trung lập

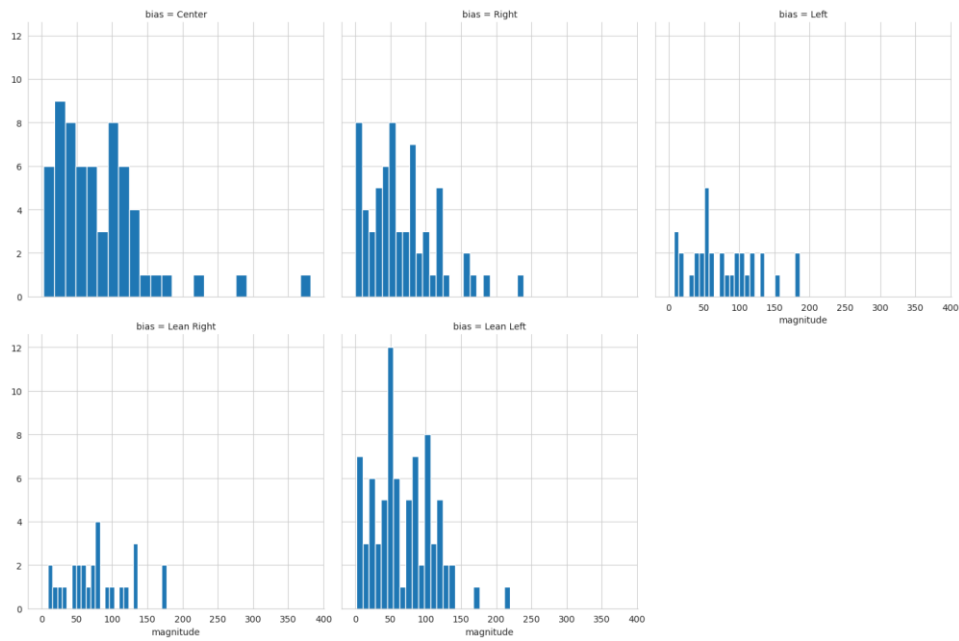
- Ta có thể thấy rằng, có khá nhiều các bài báo có cảm xúc tiêu cực có liên quan đến Donald Trump, chiếm tỉ lệ lên đến 61.89% là các bài báo có cảm xúc tiêu cực
- Các bài báo mang tính tích cực và trung lập có chưa đến một nửa (38.11%)

```
new.to_csv('news-corpus-df-sent.csv', sep='\t', encoding='utf-8')

# convert bias numbers to labels
new['bias'] = new['bias'].replace({1: 'Left', 2: 'Lean Left', 3: 'Center', 4: 'Lean Right', 5: 'Right'})
sns.set_style('whitegrid')
# visualize sentiment in relation to bias

g = sns.FacetGrid(data=new, col='bias', col_wrap= 3, height=5)
g.map(plt.hist, 'magnitude', bins=25)
plt.savefig('histogram.png', bbox_inches='tight')
```

Hình 4.2.6. Vẽ biểu đồ histogram giữa bias và magnitude

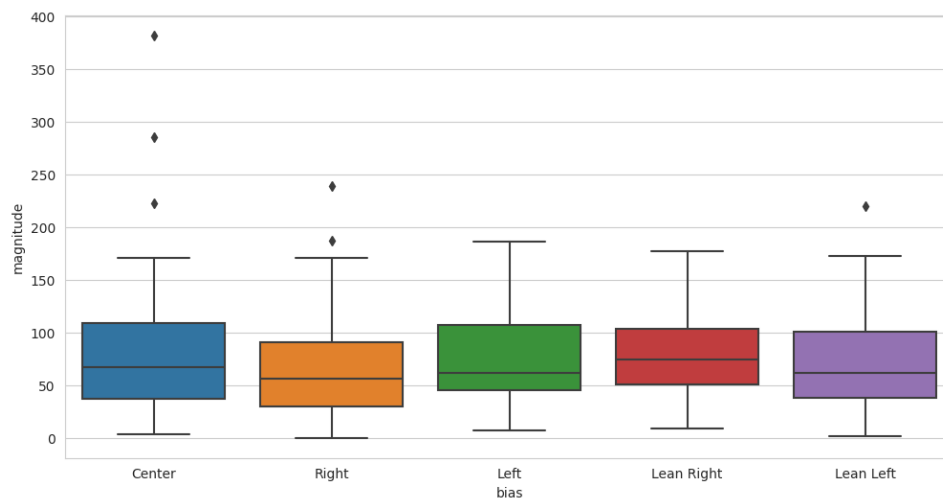


Hình 4.2.7. Biểu đồ histogram giữa bias và điểm magnitude

- Các biểu đồ histogram có độ phân phối lệch phải, tuy nhiên độ nghiêng của các biểu đồ histogram thì đối với các bài báo có tính “trung lập” có độ nghiêng lớn hơn.
- Ta có thể thấy rằng, các tờ báo đánh giá có xu hướng chính trị trung lập có các bài báo thể hiện mức độ cảm xúc của người viết bài báo rất mạnh so với 2 xu hướng cánh hữu và cánh tả.

```
plt.figure(figsize=(12,6))
sns.boxplot(x='bias', y='magnitude', data=new)
plt.savefig('boxplot.png', bbox_inches='tight')
```

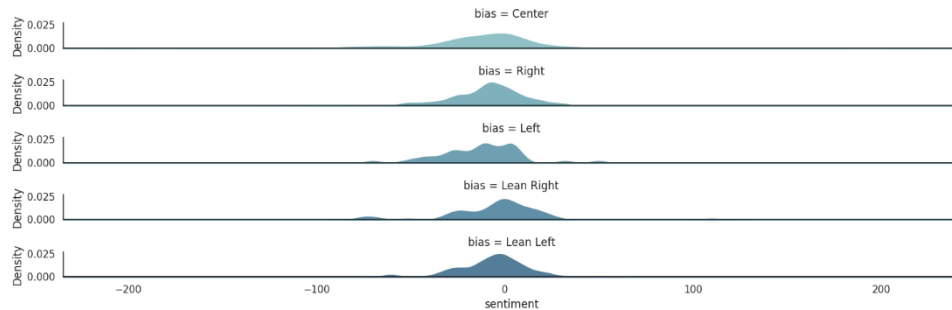
Hình 4.2.8. Vẽ boxplot





Hình 4.2.9. Biểu đồ boxplot theo từng xu hướng chính trị

- Dựa trên biểu đồ boxplot, ta có thể thấy một số điểm độ lớn cảm xúc ngoại biên (outlier) tuy nhiên đánh giá chung vẫn cho thấy rằng có các bài báo được đánh giá trung lập có xu hướng thể hiện rõ ràng cảm xúc của người viết bài báo.



Hình 4.2.10. Biểu đồ thể hiện mật độ thể hiện cảm xúc theo từng xu hướng chính trị

- Dựa vào biểu đồ mật độ trên, ta thấy có khá nhiều các bài báo tập trung ở khu vực cảm xúc trung lập, tuy nhiên, biểu đồ mật độ cũng cho thấy các bài báo có xu hướng thể hiện cảm xúc tiêu cực với số lượng nhiều hơn. Như vậy, khi tìm kiếm các bài báo có liên quan đến Donald Trump khả năng cao sẽ đọc tìm được các bài báo tiêu cực về ông ta.

### **4.3. ỨNG DỤNG TÌM KIẾM NỘI DUNG DỰA TRÊN HÌNH ẢNH (CBIR – CONTENT-BASED IMAGE RETRIEVAL)**

#### **4.3.1. XỬ LÝ SƠ BỘ CÁC ĐƯỜNG LINK**

- Load dữ liệu được cào sau khi xử lý, đánh giá cảm xúc văn bản và chọn các trường dữ liệu cần thiết

```
# create df that contains the original df
dt = pd.read_csv('/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/(SB 142023)news-corpus-df.csv')
dt = dt.loc[:,['data', 'description', 'source', 'link', 'Image_link']]

dt = dt[(dt['Image_link'] != '0')]

dt.reset_index(drop=True, inplace=True)
dt.head()
```

Hình 4.3.1.1. Bộ dữ liệu được load

|   | data                      | description                                       | source   | link  | Image_link  |
|---|---------------------------|---|----------|---|---|
| 0 | 2018-03-08T15:34:58-08:00 | The people who like the sort of tariffs that P... | cnn      | http://money.cnn.com/2018/03/05/news/economy/t... | https://i2.cdn.turner.com/money/dam/assets/180... |
| 1 | 2020-05-28T07:02:58-07:00 | President Donald Trump is preparing to sign an... | apnews   | https://apnews.com/fc30f9ebdf9d3d33a870e7374f...  | https://storage.googleapis.com/afs-prod/media/... |
| 2 | 2019-12-12T06:52:09-08:00 | The Justice Department's top watchdog on Wedne... | politico | https://www.politico.com/news/2019/12/11/horow... | https://cf-images-us-east-1-prod-boltdns.net/v... |
| 3 | 2018-07-12T12:49:56-07:00 | President Trump kept everyone guessing to the ... | wsj      | https://www.wsj.com/articles/kavanaugh-for-the... | https://s.wsj.net/img/meta/wsj-social-share.png   |
| 4 | 2019-10-17T06:20:50-07:00 | As President Donald Trump continues to fill hi... | go       | https://abcnews.go.com/Politics/exclusive-hidi... | https://s.abcnews.com/images/US/Biden-intervie... |

Hình 4.3.1.2. Bộ dữ liệu

- Xóa các hàng khỏi dataframe “new” có liên kết hình ảnh không thể truy cập bằng cách gửi yêu cầu HEAD tới từng liên kết và kiểm tra mã trạng thái phản hồi. Các hàng có mã trạng thái từ 400 trở lên sẽ bị xóa. Nếu một ngoại lệ xảy ra trong quá trình yêu cầu, hàng tương ứng cũng bị xóa.

```
import requests

# Remove rows with inaccessible image links
indices_to_remove = []
for index, row in dt.iterrows():
    url = row['Image_link']
    try:
        response = requests.head(url, allow_redirects=False)
        if response.status_code >= 400:
            indices_to_remove.append(index)
            print(f"Dropped row {index} due to error {response.status_code}")
    except requests.exceptions.RequestException as e:
        indices_to_remove.append(index)
        print(f"Dropped row {index} due to exception {e}")
dt.drop(indices_to_remove, inplace=True)
dt.reset_index(drop=True, inplace=True)
```

Hình 4.3.1.2 Kiểm tra các link image xem còn hoạt động không, nếu không còn hoạt động hay bị lỗi thì bỏ các link đó

- Xem lại bộ dữ liệu sau khi bỏ đi các dòng có đường link không truy cập được

```
dt.reset_index(drop=True, inplace=True)
dt.shape[0]

256
```

Hình 4.3.1.3 Bộ dữ liệu sau khi drop các link image bị lỗi/không còn hoạt động

#### 4.3.2. THỰC HIỆN CBIR

- Tải mô hình VGG16 được huấn luyện sẵn và tạo hàm extract\_features với input là đường dẫn hình ảnh, tải xuống và xử lý trước hình ảnh, đồng thời trích xuất các thuộc tính hình ảnh nhờ vào mô hình VGG16. Nếu function không thể xử lý ảnh, nó sẽ in thông báo lỗi và trả về kết quả None.

```
# Load the pre-trained VGG16 model
model = VGG16(weights='imagenet', include_top=False)

# Define a function to extract features from an image using the VGG16 model
def extract_features(img_path):
    try:
        response = requests.get(img_path)
        img = load_img(BytesIO(response.content), target_size=(224, 224))
        x = img_to_array(img)
        x = np.expand_dims(x, axis=0)
        x = preprocess_input(x)
        features = model.predict(x)
        return features.flatten()
    except:
        print(f"Unable to process image: {img_path}")
        return None
```

Hình 4.3.2.1. Tải mô hình VGG16 và tạo hàm extract\_features

- Tạo vòng lặp qua từng liên kết hình ảnh trong dataframe (dt), gọi hàm extract\_features để lấy các tính năng hình ảnh bằng mô hình VGG16 được đào tạo trước và lưu trữ các thuộc tính và liên kết hình ảnh trong các list riêng biệt. Nếu không thể xử lý một liên kết hình ảnh, index của hình ảnh đó sẽ được thêm vào list indices\_to\_remove.

```
# Extract features from all images in your dataset and store them in a numpy array
image_features = []
image_links = []
indices_to_remove = []

for i, img_path in enumerate(dt['Image_link']):
    features = extract_features(img_path)
    if features is not None:
        image_features.append(features)
        image_links.append(img_path)
    else:
        indices_to_remove.append(i)
```

Hình 4.3.2.2. Tách các thuộc tính từ các hình ảnh có trong bộ dữ liệu và lưu các thuộc tính

- Xóa các liên kết hình ảnh không thể truy cập khỏi dataframe, xóa các thuộc tính hình ảnh không thể truy cập tương ứng khỏi image\_features và in các chỉ số của các hình ảnh giống nhau nhất cho mỗi hình ảnh trong bộ dữ liệu được cập nhật bằng thuật toán K-Nearest Neighbors.

```

# Remove inaccessible image links from dt dataframe
indices_to_remove = [i for i in indices_to_remove if i < len(dt)]
dt.drop(dt.index[indices_to_remove], inplace=True)
dt.reset_index(drop=True, inplace=True)

# Update indices_to_remove based on the new length of dt
indices_to_remove = [i for i in indices_to_remove if i < len(dt)]

# Remove inaccessible image features from image_features array
image_features = np.array(image_features)
image_features = np.delete(image_features, indices_to_remove, axis=0)

# Print the indices of the most similar images for each image in your dataset
nbrs = NearestNeighbors(n_neighbors=3, algorithm='auto').fit(image_features)
distances, indices = nbrs.kneighbors(image_features)
for i in range(len(indices)):
    print(f"Similar images for {dt.loc[i, 'Image_link']}:")
    print(dt.loc[indices[i], 'Image_link'])

```

Hình 4.3.2.3. Xóa liên kết và thuộc tính hình ảnh, in kết quả các hình ảnh giống nhau

- Kiểm tra lại một lần nữa để xem có bị chênh lệch dữ liệu giữa bộ dữ liệu gốc và số hình ảnh đã tìm được hình ảnh giống nhau (xử lý được) ta thấy có 2 hình ảnh không xử lý được vì một lý do nào đó nên ta loại bỏ nó khỏi bộ dữ liệu gốc

```

missing_indices = set(dt.index) - set(d.index)
print(missing_indices)

{252, 253}

dt.drop(missing_indices, inplace=True)
dt.reset_index(drop=True, inplace=True)

d.shape[0]

252

dt.shape[0]

252

dt = dt.join(d)

```

Hình 4.3.2.4. Loại bỏ hình ảnh không xử lý được khỏi bộ dữ liệu gốc

- Lưu DataFrame đã cập nhật vào tệp CSV và tải hình ảnh từ các URL đã cho vào một thư mục cục bộ. Tạo vòng lặp qua DataFrame, tải từng hình ảnh xuống một đường dẫn cụ thể và tải xuống các hình ảnh tương tự cho từng hình ảnh trong DataFrame, như được biểu thị bằng cột 'similar\_images'. Đồng thời xử lý các lỗi có thể xảy ra trong quá trình tải xuống hình ảnh.

```
# Save the updated DataFrame to a file
dt.to_csv('/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/analyzing_and_same_images.csv', index=False)

# Create a directory to store the downloaded images
if not os.path.exists('/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/Images'):
    os.makedirs('/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/Images')

import urllib

for i in range(len(d)):
    image_url = dt.loc[i, 'Image_link']
    similar_indices = dt.loc[i, 'similar_images']

    # Download the image
    image_path = f"/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/Images/image_{i}.jpg"
    try:
        urllib.request.urlretrieve(image_url, image_path)
    except:
        print(f"Error downloading image {image_url}")
        continue
```

Hình 4.3.2.5. Tải hình ảnh

```
# Download the similar images
for j in similar_indices:
    similar_image_url = dt.loc[j, 'Image_link']
    similar_image_path = f"/content/drive/MyDrive/Colab Notebooks/Big Data/News_Analysis/New/Images/image_{i}_similar_{j}.jpg"
    try:
        urllib.request.urlretrieve(similar_image_url, similar_image_path)
    except:
        print(f"Error downloading similar image {similar_image_url}")
        continue
```

Hình 4.3.2.6. Tải hình ảnh giống nhau

## **4.4. ĐÁNH GIÁ CẢM XÚC HÌNH ẢNH**

### **4.4.1. XỬ LÝ CÁC ĐƯỜNG LINK**

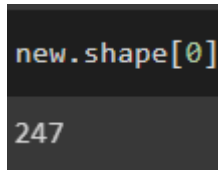
```
new = new[(new['Image_link'] != '0')]
new.reset_index(drop=True, inplace=True)
new.head()
```

Hình 4.4.1.1. Drop các dòng có đường link không có dữ liệu (giá trị "0")

- Vì một số source\_link không thể truy cập hay có thể link đã bị thay đổi domain nên một số link bài viết sẽ không truy xuất được link ảnh vậy nên kết quả image\_link sẽ trả về 0. Ta cần loại bỏ những link ảnh chỉ chứa 0 đó.
- Kết quả sau khi xử lý:

|   | main_headline  | bias   | text   | text_len | source  | Image_link   | sentiment | magnitude | senti    |
|---|--|--------|--|----------|---------|--|-----------|-----------|----------|
| 0 | kavanaugh court                                      | Center | judg brett kavanaugh speak<br>nomin presid donald... | 1062     | wsj     | https://s.wsj.net/img/meta/wsj-social-share.png        | 4.0       | 6.0       | neutral  |
| 1 | exclus hunter biden hit back trump<br>taunt exclu... | Center | presid donald trump continu fill<br>twitter feed ... | 15428    | go      | https://s.abcnews.com/images/US/Biden-<br>interview... | 59.0      | 171.0     | positive |
| 2 | republican rep cite anniversari<br>critic trump d... | Center | washington cnn two republican<br>member congress ... | 3281     | cnn     | https://media.cnn.com/api/v1/images/stellar/pr...      | -14.0     | 40.0      | negative |
| 3 | brazilian presid demand apolog<br>accept help fig... | Center | brazilian presid jair bolsonaro<br>tuesday accept... | 4322     | nbcnews | https://media-cdn.nry.s-<br>nbcnews.com/image/uploa... | 8.0       | 82.0      | positive |
| 4 | amazon fire exactli burn earth<br>lung expert say    | Center | earth lung fire burn version<br>politician journa... | 2208     | foxnews | https://static.foxnews.com/foxnews.com/content...      | -16.0     | 29.0      | negative |

Hình 4.4.1.2. Sau khi xử lý, dữ liệu giảm từ 252 dòng xuống 247 dòng.



```
new.shape[0]
247
```

Hình 4.4.1.3. Số dòng của dataframe new mới

- Một số link trong source\_link mà thư viện requests không thể truy cập sẽ gây ra lỗi vậy nên ta lọc bỏ các link đó ( trường hợp không thể truy cập thì thường thư viện requests sẽ trả về lỗi error với code > 400 . Như 401, 402, 403 ,404...)

```
# Remove rows with inaccessible image links
indices_to_remove = []
for index, row in new.iterrows():
    url = row['Image_link']
    try:
        response = requests.head(url, allow_redirects=False)
        if response.status_code >= 400:
            indices_to_remove.append(index)
            print(f"Dropped row {index} due to error {response.status_code}")
    except requests.exceptions.RequestException as e:
        indices_to_remove.append(index)
        print(f"Dropped row {index} due to exception {e}")
new = new.loc[~new.index.isin(indices_to_remove)]
new.reset_index(drop=True, inplace=True)
```

Hình 4.4.1.4. Xử lý lại lần nữa các link image không truy cập được.

- Duyệt lại một lần nữa các image\_link , với mã truy cập 200 những link không được định dạng là image sẽ bị lược bỏ đồng thời những link có định dạng hình ảnh không đọc được cũng sẽ bị lọc đi.

```

indices_to_remove = []
for index, row in new.iterrows():
    url = row['Image_link']
    try:
        response = requests.get(url)
        if response.status_code == 200:
            content_type = response.headers['Content-Type']
            if 'image' not in content_type:
                indices_to_remove.append(index)
                print(f"Dropped row {index} - {url} is not an image")
            else:
                img_data = response.content
                if img_data is None:
                    indices_to_remove.append(index)
                    print(f"Dropped row {index} - {url} returned None type image")
                else:
                    indices_to_remove.append(index)
                    print(f"Dropped row {index} - {url} returned status code {response.status_code}")
        except requests.exceptions.RequestException as e:
            indices_to_remove.append(index)
            print(f"Dropped row {index} - {e}")
    new = new.loc[~new.index.isin(indices_to_remove)]
    new.reset_index(drop=True, inplace=True)

```

Hình 4.4.1.5. Lọc lại một lần nữa các link không truy cập được cũng như các hình ảnh không sử dụng được

#### 4.4.2. ĐÁNH GIÁ CẢM XÚC HÌNH ẢNH

- Tạo thư mục images sau đó khởi tạo hàm download\_image trong đó hàm requests truy cập vào từng url để tải xuống hình ảnh . Nếu tải không thành công ảnh sẽ trả về None

```

# Create a directory to store the images
if not os.path.exists("images"):
    os.mkdir("images")

# Define a function to download and save images locally
def download_image(url, folder):
    response = requests.get(url, stream=True)
    if response.status_code == 200:
        # Extract filename from URL
        filename = os.path.join(folder, os.path.basename(url))
        # Save image to file
        with open(filename, 'wb') as f:
            f.write(response.content)
        return filename
    else:
        return None

```

Hình 4.4.2.1 Tạo thư mục images và tiến hành tải image và lưu nó vào thư mục images.

- Lấy link từ cột image\_link sau đó truyền vào hàm download để tiến hành tải các tấm ảnh xuống . Với những tấm ảnh được trả về là None thì ta loại bỏ hàng ở tệp data New

```

# Create a new column in the DataFrame for the local image path
new['Image_path'] = ''

# Download and save images
for index, row in new.iterrows():
    img_url = row['Image_link']
    try:
        response = requests.get(img_url, stream=True)
        if response.status_code == 200:
            img_path = download_image(img_url, 'images')
            if img_path is not None:
                new.at[index, 'Image_path'] = img_path
            else:
                print(f"Failed to download image at index {index}: {img_url}")
                new.drop(index, inplace=True)
        else:
            print(f"Failed to access image at index {index}: {img_url}")
            new.drop(index, inplace=True)
    except:
        print(f"Failed to access image at index {index}: {img_url}")
        new.drop(index, inplace=True)

```

Hình 4.4.2.2 Tiến hành việc tải các image và lưu nó vào thư mục images.

- Xóa các hàng khi link không thể tải xuống hình ảnh

```

# Remove rows where image download failed
new = new[new['Image_path'] != '']

# Remove corrupted images
for index, row in new.iterrows():
    img_path = row['Image_path']
    try:
        img = Image.open(img_path)
        if img is None:
            print("Removing corrupted image: ", img_path)
            os.remove(img_path)
            new.drop(index, inplace=True)
    except:
        print("Removing corrupted image: ", img_path)
        os.remove(img_path)
        new.drop(index, inplace=True)

new.dropna(inplace=True)
new.reset_index(drop=True, inplace=True)

```

Hình 4.4.2.3 Xóa các link images khi không có image nào được tải về thư mục images.

- Kết quả sau khi tải ảnh:



|   | main_headline                                     | bias   | text   | text_len | source  | Image_link  | sentiment | magnitude | sentl    | Image_path  |
|---|---|--------|--|----------|---------|---|-----------|-----------|----------|---|
| 0 | kavanaugh court                                   | Center | judge brett kavanaugh speak nomin presid donald... | 1062     | wsj     | https://s.wsj.net/img/meta/wsj-social-share.png   | 4.0       | 6.0       | neutral  | images/wsj-social-share.png                       |
| 1 | exclus hunter biden hit back trump taunt exclu... | Center | presid donald trump continu fill twitter feed...   | 15428    | go      | https://s.abcnews.com/images/US/Biden-intervie... | 59.0      | 171.0     | positive | images/Biden-interview-5577c-abc-ps-191014_hpM... |
| 2 | republican rep cite annivers critc trump d...     | Center | washington cnn two republican member congress...   | 3281     | cnn     | https://media.cnn.com/api/v1/images/stellar/pr... | -14.0     | 40.0      | negative | images/lw_800                                     |
| 3 | brazilian presid demand apolog accept help fig... | Center | brazilian presid jair bolsonaro tuesday accept...  | 4322     | nbcnews | https://media-cldnry.s-nbcnews.com/image/uploa... | 8.0       | 82.0      | positive | images/190827-emmanuel-macron-jair-bolsonaro-c... |
| 4 | amazon fire exactli burn earth lung expert say    | Center | earth lung fire burn version politician journa...  | 2208     | foxnews | https://static.foxnews.com/foxnews.com/content... | -16.0     | 28.0      | negative | images/AP-amazon-fire.jpg                         |

Hình 4.4.2.4 Bộ dữ liệu sau khi loại bỏ hết các dòng có đường link không truy cập được và không tải ảnh được

- Tạo lớp Imagedataset với dữ liệu truyền vào là ảnh sau đó biến đổi rồi trả về dưới dạng tensor

```
# Define a custom dataset class
class ImageDataset(Dataset):
    def __init__(self, df, transform=None):
        self.data = df
        self.transform = transform

    def __len__(self):
        return len(self.data)

    def __getitem__(self, idx):
        img_path = self.data.iloc[idx]["Image_path"]
        image = Image.open(img_path).convert('RGB')
        if self.transform:
            image = self.transform(image)
        return image, idx
```

Hình 4.4.2.5 Tạo class ImageDataset để đọc hình ảnh từ dataframe và trả chúng về dưới dạng tensor.

```
class ResNet18(nn.Module):
    def __init__(self, num_classes=1):
        super(ResNet18, self).__init__()
        self.resnet = torch.hub.load('pytorch/vision:v0.9.0', 'resnet18', pretrained=True)
        self.resnet.fc = nn.Linear(512, num_classes)

    def forward(self, x):
        x = self.resnet(x)
        return x
```

Hình 4.4.2.6. Tạo định nghĩa module Pytorch cho mô hình ResNet18

transforms.Resize(256): thay đổi kích thước ảnh thành kích thước 256x256.

`transforms.CenterCrop(224)`: cắt ảnh ở giữa để có kích thước 224x224.

`transforms.ToTensor()`: chuyển đổi ảnh thành một tensor PyTorch.

`transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])`: chuẩn hóa ảnh sử dụng các giá trị trung bình và độ lệch chuẩn được chỉ định trước.

```
# Define the transforms
transform = transforms.Compose([
    transforms.Resize(256),
    transforms.CenterCrop(224),
    transforms.ToTensor(),
    transforms.Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])
])
```

Hình 4.4.2.7. Sử dụng lớp Compose trong Pytorch để tạo một chuỗi các chuyển đổi hình ảnh

Chuẩn bị môi trường cho việc sử dụng mô hình ResNet18 để dự đoán cảm xúc trên các hình ảnh đã tải xuống và lưu trữ trong new

```
# Define the dataset
dataset = ImageDataset(new, transform=transform)

# Define the dataloader
dataloader = DataLoader(dataset, batch_size=1, shuffle=False)

# Define the device to use for training
device = torch.device("cuda:0" if torch.cuda.is_available() else "cpu")

# Create an instance of the ResNet18 model
model = ResNet18(num_classes=1).to(device)

# Set the model to evaluation mode
model.eval()
```

Hình 4.4.2.8. thiết lập môi trường sử dụng mô hình ResNet18 để dự đoán cảm xúc của hình ảnh trong tập dữ liệu

- Lặp qua từng data batch trong dataloader, di chuyển dữ liệu đến thiết bị được chỉ định (GPU hoặc CPU), chuyển dữ liệu qua mô hình để nhận đầu ra, áp dụng function sigmoid cho đầu ra và chuyển đổi tenxơ đầu ra thành một mảng numpy để trích xuất điểm số cảm xúc được dự đoán bởi mô hình. Các điểm số dự đoán này sau đó được thêm vào list prediction và các chỉ số tương ứng của hình ảnh trong bộ dữ liệu gốc cũng được ghi lại trong list indices.

```
# Evaluate the model on the dataset
predictions = []
indices = []
with torch.no_grad():
    for data, idx in dataloader:
        data = data.to(device)
        output = model(data)
        prediction = torch.sigmoid(output).cpu().detach().numpy()[0][0]
        predictions.append(prediction)
        indices.append(idx)
```

Hình 4.4.2.9. Đánh giá mô hình được huấn luyện trên bộ dữ liệu hình ảnh và đưa ra dự đoán cảm xúc

- Sau khi đã xong việc đánh giá cảm xúc hình ảnh, tiến hành tách kết quả đánh giá cảm xúc với tên cột là “Sentiment” và lấy lại cột “Image\_path” tương ứng đưa vào dataframe tên “results”. Sau đó lưu cột “Sentiment” của “results” vào một dataframe mới “dataset.data” với tên cột tương ứng

```
# Create a new DataFrame with the original data and the predictions
results = pd.DataFrame({
    "Image_path": dataset.data["Image_path"],
    "Sentiment": predictions
})

# Add the predictions to the original DataFrame
dataset.data["Sentiment"] = results["Sentiment"]
```

Hình 4.4.2.10. Đưa kết quả vào một dataframe

- Đọc dữ liệu của dataframe “results”

```
results.head()
```

|   | Image_path  | Sentiment |
|---|---|-----------|
| 0 | images/wsj-social-share.png                       | 0.550776  |
| 1 | images/Biden-interview-5577c-abc-ps-191014_hpM... | 0.630898  |
| 2 | images/w_800                                      | 0.374618  |
| 3 | images/190827-emmanuel-macron-jair-bolsonaro-c... | 0.609227  |
| 4 | images/AP-amazon-fire.jpg                         | 0.534967  |

Hình 4.4.2.11. Dataframe chứa kết quả phân tích cảm xúc hình ảnh.

- Gắn nhãn cho từng hình ảnh dựa trên số điểm đánh giá cảm xúc mô hình đưa ra:
  - Điểm Sentiment lớn hơn 0.5 là “positive”

- Điểm Sentiment nhỏ hơn 0.5 là “negative”
- Điểm Sentiment 0.5 là “neutral”

```
for i, row in dataset.data.iterrows():
    if row["Sentiment"] > 0.5:
        dataset.data.at[i, "Sentiment"] = "positive"
    elif row["Sentiment"] < 0.5:
        dataset.data.at[i, "Sentiment"] = "negative"
    else:
        dataset.data.at[i, "Sentiment"] = "neutral"
```

Hình 4.4.2.10. Thực hiện việc gán nhãn cho từng ảnh dựa trên số điểm sentiment

#### 4.4.3 KẾT QUẢ THỰC HIỆN

|   | main_headline  | bias   | text   | text_len | source  | Image_link  | sentiment | magnitude | senti    | Image_path  | Sentiment |
|---|--|--------|--|----------|---------|---|-----------|-----------|----------|---|-----------|
| 0 | kavanaugh court  | Center | judy breitt<br>kavanaugh<br>speak nomin<br>presid<br>donald... | 1062     | wsj     | https://s.wsj.net/img/meta/wsj-social-share.png   | 4.0       | 6.0       | neutral  | images/wsj-social-share.png                       | positive  |
| 1 | exclus hunter<br>biden hit back<br>trump taunt<br>exclu... | Center | presid donald<br>trump continu<br>fill twitter feed<br>...     | 15428    | go      | https://s.abcnews.com/images/US/Biden-intervie... | 59.0      | 171.0     | positive | images/Biden-interview-5577c-abc-ps-191014_hpM... | positive  |
| 2 | republican rep cite<br>anniversari critic<br>trump d...    | Center | washington<br>cnn two<br>republican<br>member<br>congress ...  | 3281     | cnn     | https://media.cnn.com/api/v1/images/stellar/pr... | -14.0     | 40.0      | negative | images/w_800                                      | neutral   |
| 3 | brazilian presid<br>demand apolog<br>accept help fig...    | Center | brazilian<br>presid jair<br>bolsonaro<br>tuesday<br>accept...  | 4322     | nbcnews | https://media-cldnry.s-nbcnews.com/image/uploa... | 8.0       | 82.0      | positive | images/190827-emmanuel-macron-jair-bolsonaro-c... | positive  |
| 4 | amazon fire exactli<br>burn earth lung<br>expert say       | Center | earth lung fire<br>burn version<br>politician<br>journa...     | 2208     | foxnews | https://static.foxnews.com/foxnews.com/content... | -16.0     | 28.0      | negative | images/AP-amazon-fire.jpg                         | positive  |

Hình 4.4.3.1. Bộ dữ liệu sau khi hoàn tất việc phân tích cảm xúc hình ảnh

- Nhìn sơ bộ kết quả đánh giá cảm xúc hình ảnh có sự chênh lệch tương đối lớn so với kết quả đánh giá cảm xúc văn bản. Điều đáng lý không nên diễn ra, tuy nhiên, lý do là do nhóm đang sử dụng mô hình được huấn luyện sẵn với file điểm trọng số đi kèm. Do đó kết quả không được tối ưu hóa cho bộ dữ liệu của nhóm.

```
[ ] # Save the updated DataFrame
dataset.data.to_csv('/content/drive/MyDrive/Colab Notebooks/Big Data/News Analysis/New/image_sentiment.csv', index=False)
```

Hình 4.4.3.2. Lưu file dữ liệu kết quả

## CHƯƠNG 5. KẾT LUẬN

### 5.1. CÁC KẾT QUẢ ĐẠT ĐƯỢC

- Nhóm đã thành công trong việc cào dữ liệu các bài báo từ web allsides.
- Thực hiện được việc phân tích cảm xúc của từng bài báo sử dụng mô hình học máy.
- Thực hiện được việc tìm kiếm các hình ảnh giống nhau dựa trên các dòng hình ảnh, đây là tiền đề cho việc phát triển một hệ thống CBIR hoàn chỉnh.
- Thực hiện được việc phân tích cảm xúc của các hình ảnh liên quan đến bài báo sử dụng mô hình học sâu và thị giác máy tính.

### 5.2. NHỮNG HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN

#### ➤ Những hạn chế:

- Đối với việc tiền xử lý để phân tích cảm xúc bài báo còn chưa hợp lý vì có một vài từ không ở đúng dạng chuẩn của nó.
- Các link url của bài báo đôi khi bị chết, bị chặn truy cập hoặc bị thay đổi nên sẽ có nhiều bài báo không cào được thông tin.
- Thiết bị của sinh viên có giới hạn nên data không đủ độ lớn để tiến hành training qua đó độ chính xác của kết quả không cao
- Chưa tối ưu hóa, tự động hóa được việc cào dữ liệu từ web allsides, tuy việc thực hiện cào dữ liệu của nhóm rất công phu nhưng lại được thực hiện phần lớn là thủ công.
- CBIR vẫn chưa tối ưu hóa, có rất nhiều các hình ảnh giống nhau trùng lặp cho loại bỏ được.
- Không đủ khả năng về thiết bị và thời gian để huấn luyện được mô hình riêng biệt cho việc thực hiện CBIR và đánh giá cảm xúc hình ảnh. Dẫn đến kết quả không được tốt như mong đợi

#### ➤ Hướng phát triển:

- Tự động hóa việc cào dữ liệu từ trang AllSides.com và giảm thiểu hoàn toàn việc cào dữ liệu thủ công.
- Huấn luyện lại mô hình đã được huấn luyện trước dựa trên tập dữ liệu của nhóm để tối ưu hóa CBIR và đánh giá cảm xúc hình ảnh.
- Sau khi xây dựng hoàn tất mô hình Image Sentiment ta có thể đánh giá một bài báo thông qua tấm ảnh bìa được đăng kèm.
- Thay đổi các thông số của mô hình để tối ưu hóa mô hình.
- Tạo trang web tự động cào dữ liệu, phân tích cảm xúc văn bản và hình ảnh
- Image Sentiment Analysis chủ yếu tập trung vào phân tích các hình ảnh tĩnh. Tuy nhiên, với sự phát triển của công nghệ, việc phân tích cảm xúc trong hình ảnh động cũng đang được nghiên cứu và phát triển. **Hướng tới phân tích cảm xúc video.**

# TÀI LIỆU THAM KHẢO VÀ CÔNG CỤ HỖ TRỢ

## **TÀI LIỆU THAM KHẢO:**

<https://telehub.vn/tinh-nang-tong-dai/sentiment-analysis/>

[Image Sentiment Analysis Using Deep Learning | IEEE Conference Publication | IEEE Xplore](#)

N. Mittal, D. Sharma and M. L. Joshi, "Image Sentiment Analysis Using Deep Learning," 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), Santiago, Chile, 2018, pp. 684-687, doi: 10.1109/WI.2018.00-11.

Content Based Image Retrieval System (Mohd Omar (Maulana Azad National Urdu University, India), Khaleel Ahmad (Maulana Azad National Urdu University, India), and M.A. Rizvi (NITTTR, India))

<https://github.com/thisandagain/sentiment>

<https://www.freecodecamp.org/news/what-is-sentiment-analysis-a-complete-guide-to-for-beginners/>

[https://www.researchgate.net/publication/338916787\\_Survey\\_on\\_Visual\\_Sentiment\\_Analysis](https://www.researchgate.net/publication/338916787_Survey_on_Visual_Sentiment_Analysis)

## **CÔNG CỤ HỖ TRỢ:**

ChatGPT: chat.openai.com

Bing AI: bing.com/new

BARD AI: bard.google.com