

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG



Môn học: Lưu Trữ Và Xử Lý Dữ Liệu Lớn

Đề tài: Lưu trữ và xử lý dữ liệu sách trên Vinabook

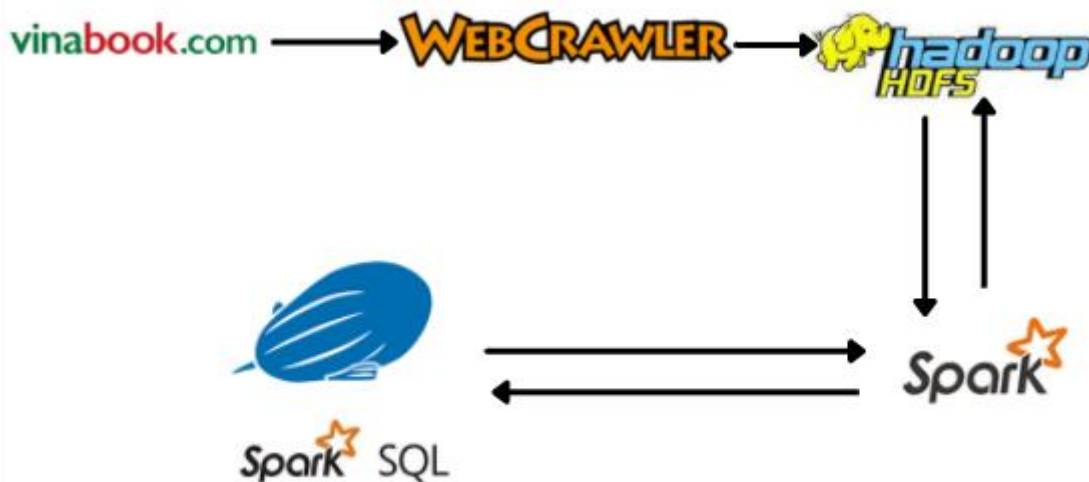
Lớp : 128751
Giáo viên hướng dẫn : TS. ĐÀO THÀNH CHUNG
Nhóm sinh viên : LoveData

Họ và tên	MSSV
Phương Trung Kiên	20183776
Đào Trọng Hiếu	20183739
HIM PHOM	20170283
Lê Nguyễn Thành Chung	20183696

Hà Nội, tháng 12 năm 2021

CHƯƠNG 1. Đặt vấn đề bài toán

Dưới sức ép của sự bùng nổ công nghệ thông tin, cuộc sống của con người đang chuyển mình để thích ứng với thời cuộc. Nổi bật lên trong đó là sự thay thế của hàng loạt cửa hàng truyền thống bằng cửa hàng online, trong đó có những cửa hàng sách online, điển hình như Vinabook, Nhã Nam, Tiki, ... Bên cạnh nhu cầu sắp xếp và quản lý một lượng lớn sách thì những cửa hàng này cũng cần thực hiện phân loại những quyển sách này vào đúng topic của chúng. Việc lưu trữ cũng như phân loại này nếu thực hiện bằng tay sẽ rất tốn kém nguồn lực và tài lực. Nhận thấy tác vụ này có thể giải quyết được khi sử dụng lưu trữ và xử lý dữ liệu lớn, nhóm LoveData quyết định triển khai một mô hình có thể lưu trữ xử lý dữ liệu sách để có thể thuận tiện hơn trong việc sử dụng chúng.



CHƯƠNG 2. Thu thập dữ liệu

Dữ liệu là đối tượng không thể thiếu trong Big Data. Độ chính xác của hoạt động được dự đoán là phụ thuộc đến 70% vào số lượng và chất lượng dữ liệu thu thập được. Trên thực tế rất khó để thu thập được bộ dữ liệu tốt (thường được gọi là dataset). Dữ liệu có thể chứa đầy nhiễu và các giá trị thiếu. Các bộ dữ liệu tốt quan trọng đến mức có thể hơn cả thuật toán tốt. Vì vậy, tìm được nguồn dữ liệu để có thể thu thập đủ và tốt là điều cực kỳ quan trọng.

2.1. Nguồn dữ liệu

Mô hình đưa ra là phân loại sách vào đúng topic của chúng, vì vậy nhóm đã tham khảo một vài website chứa thông tin sách và đi đến kết luận sử dụng website <https://www.vinabook.com>. Lý do nhóm quyết định chọn thu thập dữ liệu từ trang web này vì số lượng sách lên tới 100 000 quyển cùng với các topic đa dạng và phong phú, phù hợp với mục đích của đề tài.

2.2. Phương pháp thu thập dữ liệu

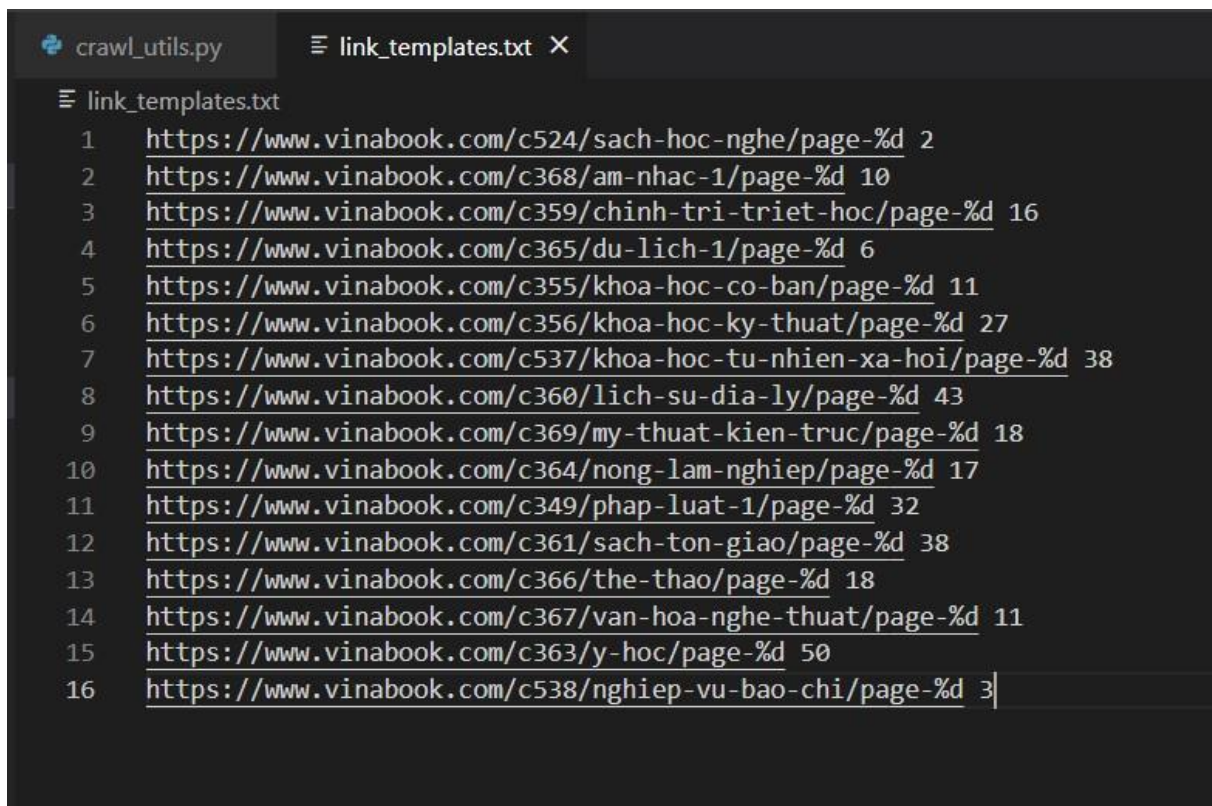
Nhóm sử dụng phương pháp Web Crawling là quá trình tự động trích xuất các thông tin từ trang web và lưu trữ nó dưới một định dạng phù hợp. Chương trình mà công việc này thực hiện gọi là web crawler. Thông thường, khi muốn lấy một số thông tin từ trang web, mọi người sẽ dùng các API mà các trang đó cung cấp. Tuy nhiên không phải trang web nào cũng cung cấp sẵn API để chúng ta sử dụng. Vì vậy ta cần một kỹ thuật khác để lấy các thông tin từ trang web mà không cần API, đó là sử dụng thư viện BeautifulSoup. Thư viện BeautifulSoup là một thư viện của Python cho phép lấy dữ liệu từ HTML đơn giản và hiệu quả.

2.2.1. Các bước thu thập dữ liệu

Các bước thu thập dữ liệu bao gồm :

- Chuẩn bị link topic sách
- Crawl links sách theo topic
- Crawl nội dung của sách

Link topic của sách chuẩn bị thủ công bao gồm các danh mục sách bao gồm tất cả các đầu mục ở trang web <https://www.vinabook.com> như âm nhạc, chính trị triết học, du lịch, khoa học cơ bản, khoa học kỹ thuật, ... Tiếp theo nhóm sẽ thu thập tất cả các link sách theo từng danh mục, cuối cùng crawl dữ liệu của từng sách.



```
crawl_utils.py  link_templates.txt X
link_templates.txt
1 https://www.vinabook.com/c524/sach-hoc-nghe/page-%d 2
2 https://www.vinabook.com/c368/am-nhac-1/page-%d 10
3 https://www.vinabook.com/c359/chinh-tri-triet-hoc/page-%d 16
4 https://www.vinabook.com/c365/du-lich-1/page-%d 6
5 https://www.vinabook.com/c355/khoa-hoc-co-ban/page-%d 11
6 https://www.vinabook.com/c356/khoa-hoc-ky-thuat/page-%d 27
7 https://www.vinabook.com/c537/khoa-hoc-tu-nhien-xa-hoi/page-%d 38
8 https://www.vinabook.com/c360/lich-su-dia-ly/page-%d 43
9 https://www.vinabook.com/c369/my-thuat-kien-truc/page-%d 18
10 https://www.vinabook.com/c364/nong-lam-nghiep/page-%d 17
11 https://www.vinabook.com/c349/phap-luat-1/page-%d 32
12 https://www.vinabook.com/c361/sach-ton-giao/page-%d 38
13 https://www.vinabook.com/c366/the-thao/page-%d 18
14 https://www.vinabook.com/c367/van-hoa-nghe-thuat/page-%d 11
15 https://www.vinabook.com/c363/y-hoc/page-%d 50
16 https://www.vinabook.com/c538/nghiep-vu-bao-chi/page-%d 3|
```

Để thu thập dữ liệu từ trang web, nhóm sử dụng hàm `request.get()`. Thư viện `request` sẽ tạo yêu cầu GET cho máy chủ web để tải nội dung HTML.

```
def get_page_content(self, page_number):  
    url = self.get_url(page_number)  
    response = requests.get(url, headers={'User-Agent': 'Mozilla/5.0'})  
    assert response.status_code == 200, "Can't connect to this url %s" % url  
    return bs4.BeautifulSoup(response.text, 'html.parser')
```

Sau đó nhóm sẽ trích xuất dữ liệu thô của trang như : link sách, tên sách, giới thiệu, tác giả, nhà xuất bản, ngôn ngữ, ... Các trường dữ liệu trên đều được viết dưới dạng các tag, để lấy được các trường thông tin trong đó, nhóm sử dụng thư viện BeautifulSoup.

```
# name  
try:  
    name = soup.find('h1', {'itemprop': 'name'}).text.strip()  
except:  
    name = None  
book_info['name'] = name
```

Theo ảnh trên nhóm đã tìm được nội dung của thẻ `<h1>` là tên sách. Như vậy nhóm đã lấy được các trường thông tin của sách từ một tag, sau đó lưu giữ thông tin dưới vào một file csv.

2.2.2. Mô tả dữ liệu để áp dụng vào visualize

Thuộc tính của dữ liệu là yếu tố cốt lõi của bộ dữ liệu. Mỗi đối tượng có vô số đặc tính mà bài toán chỉ cần xét đến một số thuộc tính nào đó. Đây là những yếu tố mà mô hình nhìn vào và sử dụng để dự đoán.

Các trường dữ liệu của một quyển sách mà nhóm quyết định thu thập bao gồm: danh mục, lời giới thiệu, tên sách, giá bán, ảnh, tác giả, người dịch, nhà xuất bản, nhà in, id, cân nặng, ngôn ngữ, format, kích thước, ngày phát hành, số trang. Trong đó, những trường dữ liệu quan trọng nhất là lời giới thiệu, tên sách và danh mục sách. Ví dụ dưới đây mô tả các trường dữ liệu thu thập được của một quyển sách.

```
book_info['authors'] = None
book_info['translator'] = None
book_info['publisher'] = None
book_info['printer'] = None
book_info['id'] = None
book_info['weight'] = None
book_info['language'] = None
book_info['format'] = None
book_info['width'] = None
book_info['height'] = None
book_info['publish_date'] = None
book_info['pages'] = None
return book_info
```

2.3. Vấn đề gặp phải khi thu thập dữ liệu

Nhóm đã gặp phải một số khó khăn trong quá trình thu thập dữ liệu. Một vài trang web hạn chế số lượng truy cập trong một khoảng thời gian để tránh quá tải server. Khi có một số lượng lớn request trong vài phút, việc crawler có thể bị cấm. Ngoài ra, việc crawling tùy thuộc vào từng layout của trang web và layout của nó có thể thay đổi theo thời gian. Do đó khi layout thay đổi, nhóm cũng sẽ phải thay đổi code.

Trên đây là những vấn đề mà nhóm đã gặp phải trong quá trình thu thập dữ liệu, ngoài ra vẫn còn khá nhiều vấn đề mà nhóm còn chưa gặp, mong thầy có thể góp ý để nhóm cùng thảo luận. Tiếp sau khi thu thập dữ liệu, nhóm sẽ tiến hành làm xử lý dữ liệu.

CHƯƠNG 3. Làm sạch dữ liệu

xử lý dữ liệu thô là một bước rất quan trọng trong việc visuallize dữ liệu.

Đối với bài toán phân loại sách, việc tiền xử lý dữ liệu văn bản tiếng Việt là quá trình từ bộ dữ liệu thô được thu thập, tiến hành chuẩn hóa dữ liệu và loại bỏ các thành phần không có ý nghĩa cho việc phân loại văn bản. Các bước thực hiện như sau:

- Bước 1: Xóa code HTML còn dư
- Bước 2: Đưa văn bản về dạng chữ thường
- Bước 3: Xóa bỏ các ký tự không cần thiết
- Bước 4: Thực hiện tách từ tiếng Việt - Bước 5: Xóa stopwords và uniqueword

3.1. Xóa bỏ các thẻ HTML còn dư

Dữ liệu được thu thập từ các website đôi khi vẫn còn sót lại các đoạn mã HTML. Những đoạn mã HTML code này là “rác”, chẳng những không có tác dụng cho việc phân loại mà còn làm kết quả phân loại văn bản bị kém đi rất nhiều. Do đó, việc xóa các HTML là rất cần thiết. Cách thực hiện cũng khá đơn giản, chúng ta sẽ sử dụng **regex** để xử lý.

Trước hết, chúng ta cần tìm hiểu **Regular Expression** là gì. **Regular Expression (RegEx)** hay còn gọi là Biểu thức chính quy là một đoạn các ký tự đặc biệt theo những khuôn mẫu (pattern) nhất định, đại diện cho chuỗi hoặc một tập các chuỗi.

Ví dụ một pattern như sau: `^abc` → Khớp với các chuỗi bắt đầu bằng chuỗi abc

Để sử dụng Regular Expression trong Python thì cần phải import module re vào chương trình. Module re đã có sẵn trong python3 và không cần phải cài đặt thêm.

Trở lại với bài toán, để xóa bỏ các code html còn dư từ bước thu thập dữ liệu thô, ta sử dụng regex và với pattern hợp lý như ví dụ sau:

```
In [1]: 1 import re
        2
        3 def remove_html(txt):
        4     return re.sub(r'<[^>]*>', '', txt)
        5
        6 txt = "<p class=\"par\">Chuyên Đề Ôn Tập 12</p>"
        7 remove_html(txt)

Out[1]: 'Chuyên Đề Ôn Tập 12'
```

Ở đây, chuỗi định dạng mẫu được sử dụng là '`<[^>]*>`'. Chuỗi ký tự pattern này sẽ khớp với các chuỗi có định dạng bắt đầu bằng ký tự mở ngoặc nhọn `<`, kết thúc bằng ký tự `>`, trong cặp ngoặc nhọn đó là chuỗi có hoặc không có dấu đóng ngoặc (trường hợp có 2 thẻ html lồng nhau). Chuỗi pattern sẽ khớp với các thẻ html ví dụ như `<p>`, `</div>` và loại bỏ chúng đi khỏi văn bản.

3.2. Đưa văn bản về dạng chữ thường

Việc đưa dữ liệu văn bản về chữ viết thường là rất cần thiết. Máy tính hiểu chữ viết hoa thường là 2 từ khác nhau. Mà chữ in hoa không có tác dụng ở bài toán phân loại văn bản, vì 1 từ có chữ thường hay in hoa đều mang ý nghĩa như nhau. Đưa về chữ viết thường giúp đáng kể số lượng từ mà mô hình phải học và tăng độ chính xác hơn cho mô hình.

Trong python, để chuyển văn bản sang chữ thường đơn giản chỉ cần gọi hàm `document.lower()`.

```
In [3]: 1 def tolower_case(txt):
        2     return txt.lower()
        3
        4 txt = "Chuyên Đề Ôn Tập Cuối Kỳ"
        5 tolower_case(txt)

Out[3]: 'chuyên đề ôn tập cuối kỳ'
```


3.4. Tách từ tiếng Việt

Như đã biết, trong tiếng Việt chúng ta có từ đơn và từ ghép. Từ đơn là từ do một tiếng có nghĩa tạo nên (ví dụ: trường). Từ ghép là từ do hai hoặc nhiều tiếng tạo nên (ví dụ sinh viên). Với 1 đoạn văn bản thông thường, mô hình sẽ hiểu tất cả các từ đều là từ đơn, cách nhau bởi 1 dấu cách. Vậy ta phải xử lý văn bản đầu vào để mô hình hiểu được đâu là từ đơn và từ ghép. Tức là với mỗi từ ghép hiểu theo nghĩa tiếng Việt, chúng ta thêm “liên kết” để thành từ ghép mà mô hình hiểu được. Đó là bài toán tách từ (Word tokenizing) cũng là 1 bài toán khác trong lĩnh vực xử lý ngôn ngữ tự nhiên.

Ví dụ: Sinh viên đến báo cáo bài tập => Sinh_viên đến báo_cáo bài_tập

Hiện nay có khá nhiều thư viện mã nguồn mở của bài toán này. Do đó, chúng ta chỉ việc cài đặt và sử dụng chúng. Với ngôn ngữ Python, ta có các thư viện **underthesea** hoặc **pyvi** có thể xử lý tốt bài toán này.

Trở lại với bài toán, em sử dụng thư viện underthesea để tách từ tiếng Việt. Thư viện này không có sẵn trong gói cài đặt python 3. Vậy ta phải cài đặt thông qua trình quản lý thư viện pip bằng câu lệnh: **pip install underthesea**. Cách sử dụng thư viện đơn giản, theo như ví dụ dưới đây:

```
1 from underthesea import word_tokenize
2
3 def vietnamese_word_tokenize(doc):
4     return word_tokenize(doc, format="text")
5
6
7 text = "Cách làm giàu ở đây là an toàn và chắc chắn vì nó dựa trên những nguyên tắc đã được
8
9 print("Before word tokenizing:", text)
10 print('-----')
11 print("After word tokenizing:", vietnamese_word_tokenize(text))
```

Before word tokenizing: Cách làm giàu ở đây là an toàn và chắc chắn vì nó dựa trên những nguyên tắc đã được kiểm nghiệm qua thực tế, những nguyên tắc không nhằm vào việc lợi dụng hoặc hại ai khác người khác để làm lợi cho mình.

After word tokenizing: Cách làm giàu ở đây là an_toàn và chắc_chắn vì nó dựa trên những nguyên_n_tắc đã được kiểm_nghiệm qua thực_tế , những nguyên_tắc không nhằm vào việc lợi_dụng hoặc kh aỉ_thác người khác để làm_lợi cho mình .

3.5. Xóa stop words và unique words

Bên cạnh các xử lý trên, chúng ta còn phải cần quan tâm đến 1 xử lý khác nữa. Đó là việc loại bỏ **stop word** và **unique word**. Bước xử lý này góp phần giúp giảm số lượng từ, loại bỏ các từ không mang ý nghĩa, tăng tốc độ học và xử lý, và nâng cao kết quả của mô hình.

- Thứ nhất, Stopword (từ dừng) là những từ xuất hiện nhiều trong ngôn ngữ tự nhiên, tuy nhiên lại không mang nhiều ý nghĩa. Ở tiếng việt Stopwords là các từ nối (của, là, có, được, những,...). Danh sách stopwords phải được xây dựng từ bộ dữ liệu văn bản lớn, của toàn bộ bộ dữ liệu. Để loại bỏ các stopwords, ta sẽ làm theo phương pháp dựa trên tần suất xuất hiện của từ. Ta tiến hành đếm số lần xuất hiện của từng từ trong dataset. Dữ liệu về các tần suất xuất hiện các từ sẽ được lưu vào 1 cấu trúc dictionary với key là chính từ ấy, value là số lần xuất hiện. Dữ liệu sau đó được sắp xếp theo chiều giảm dần value, tức là các từ có tần suất xuất hiện càng cao thì xếp trên đầu và nhiều khả năng là các stopwords. Ta lấy top 100 (có thể lấy nhiều hoặc ít hơn) của dữ liệu tần suất vừa thu được, sẽ là các từ stopwords cần loại bỏ. Ví dụ top 100 stop words trong dataset:

```
D:\Workplaces\Github\BookTopicClassification>python test.py
Danh sách 100 từ xuất hiện nhiều nhất:
của và những trong là một người các được có
về cho với không đã sách để sự đọc đến
cũng bạn như này từ mình lại nhiều mà ra
khi đó ở phải nhưng cuốn vào hơn có thể nam
việt đi khác năm sẽ trên cuộc_sống mới chương rất
theo chúng_ta qua còn nhân_vật chính tìm thế_giới đón làm
trẻ mời cái thấy tác_giả tác_phẩm sống viết mỗi nhất
cách hay chỉ thì biết phần việc hai bệnh nước
lịch_sử nhìn darren con cùng văn ông con_người hạnh_phúc mọi
trước đang tập bộ điều thứ từng ta tôi khiến
```

- Thứ 2, unique word (từ duy nhất) là các từ xuất hiện rất ít trong văn bản, với tần suất 1-2 lần. Những từ này cũng là những từ không mang giá trị trong quá trình huấn luyện bởi vì nếu mô hình chỉ gặp nó một lần mà không gặp lại lần nào nữa thì mô hình sẽ không học được gì từ những từ đó. Phần lớn những từ này là tên riêng, số, từ tiếng anh thậm chí là những từ viết sai chính tả, vì vậy, loại bỏ unique words cũng là một cách

rất tốt để loại bỏ nhiễu (lỗi đánh máy) ra khỏi dữ liệu. Dựa trên dữ liệu về tần suất xuất hiện các từ đã tính được, ta trích xuất ra danh sách các từ xuất hiện 1 lần duy nhất (dưới 2 lần) và loại bỏ chúng. Đó chính là các từ unique word và cần loại bỏ khỏi văn bản.

Danh sách từ xuất hiện 1 lần duy nhất: (100 từ)

đêm	dân	hát	độc	giả	bộ	tình	ca	vũ	organ	bộ	bảo	mời	phong	sương	ấn	bản	nhạc	tích	cóp	dạy	kèm	xuất	sắc	chỉ
micro	phong	thanh	flute	cp90	trvn	thơ	huy	cận	truyện	hải	hòa	tỉnh	trầm	hùng	lê	văn	hào							
cuối	bản	nhạc	kịch	hào	nhạc	mời	nsut	đánh	dân	mélodie	spring	sonatina	spindler											
lay	động	lòng	nhà	twist	ballade	adeline	sonate	souvenirs	enfance	letter	ma	mere												
gammon	unchange			melody	acoustic	1817	almeria	ré	quạt	picnic	u	hoài	flamenco											
lục	anh	bằng	gánh	đặng	hiên	biển	phương	buồn	trọng	chứa	lễ	chuyến	phương	đan	ngọc	diệp	hồ	định	lục	bạn				
tôi	bén	quê	bóng	thêm	các	đích	chờ	maicon	bến	dạ	khúc	dưới	cũ	dứt	to	duyên	quê	đêm						
đồ	chân	đôi	tây	đêm	cùng	được	mùa	đường	với	4	hoang	5	xanh	6	mỹ	7	xưa	8	elvis					
giang	gửi	ngủ	mục	juliette	2	mười	3	với	4	hoang	5	xanh	6	mỹ	7	xưa	8	elvis						
presley	achy	beraky	seen	the	rain	take	handy	devoted	got	friend														

3.6. Tổng kết

Từ dữ liệu thô ban đầu của bước thu thập dữ liệu, mỗi một quyển sách có nhiều trường dữ liệu là topic, introduction, name, price, cover, authors, ... Ta tiến hành trích xuất lấy 3 trường là **name**, **introduction**, **topic**. Dữ liệu input sẽ có được bằng cách ghép **name** + **introduction** + **name**. Còn output là nhãn cho mỗi cuốn sách sẽ là trường **topic**. Vậy tại sao lại đặt trường input như vậy? Đơn giản là vì tiêu đề của mỗi cuốn sách chứa rất nhiều thông tin về chủ đề của cuốn sách đó, và những thông tin đó rất cô đặc và xúc tích, nên để tận dụng nguồn dữ liệu đã được thu thập từ trang web, chúng ta sẽ gộp trường dữ liệu name và introduction vào với nhau và tạo thành trường input. Đặc biệt, vì mật độ thông tin trong name sẽ lớn hơn so với introduction, nên nhóm em có sử dụng một kỹ thuật nhỏ để làm mô hình tập trung học phần name hơn so với phần còn lại của input. Đó là ghép 2 lần name và 1 lần introduction vào input. Bằng cách này, sau khi qua bộ trích chọn đặc trưng, những từ này sẽ có độ ảnh hưởng cao hơn tới chủ đề mà mô hình dự đoán so với phần còn lại của input, đặc biệt là với phương pháp Bag of Words.

Tiếp sau đó, dữ liệu input sẽ được đi qua các bước tiền xử lý đã trình bày bên trên để có được nguồn dữ liệu đã chuẩn hóa và đã được thu gọn, đảm bảo làm sạch, loại bỏ các thành phần không có ý nghĩa.

CHƯƠNG 4. Lưu Trữ dữ liệu

4.1. Chuẩn bị

-Để lưu trữ dữ liệu ta sẽ tạo 1 cụm HDFS nhằm phục vụ cho lưu trữ và xử lý dữ liệu một cách dễ dàng.

- Cài đặt và sử dụng phiên bản Hadoop3.2 ở trên 3 máy:

```
192.168.3.11    kienpt
192.168.3.11    nodems
192.168.3.12    node1
192.168.3.13    node2
```

In operation

Show entries

Search:




Node	Http Address	Last contact	Last Block Report	Capacity	Blocks	Block pool used	Version
✓ node1:9866 (192.168.3.12:9866)	http://node1:9864	2s	2m	19.07 GB <div><div></div></div>	20	2.05 GB (10.77%)	3.1.2
✓ node2:9866 (192.168.3.13:9866)	http://node2:9864	1s	2m	19.07 GB <div><div></div></div>	20	2.05 GB (10.77%)	3.1.2

Showing 1 to 2 of 2 entries

Previous **1** Next







4.2. Lưu trữ

-Dữ liệu sau khi crawl sẽ được đẩy thẳng vào HDFS tổng dữ liệu lưu trữ 760MB:

Go!   

Show entries

Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	-rw-r--r--	kien	supergroup	251.28 MB	Jan 03 23:59	2	128 MB	BTLbigdata.csv	
<input type="checkbox"/>	-rw-r--r--	kien	supergroup	313.16 MB	Jan 03 22:00	2	128 MB	data.csv	
<input type="checkbox"/>	-rw-r--r--	kien	supergroup	329.7 MB	Jan 03 23:51	2	128 MB	data2.csv	
<input type="checkbox"/>	-rw-r--r--	kien	supergroup	329.26 MB	Jan 04 08:05	2	128 MB	data3.csv	
<input type="checkbox"/>	-rw-r--r--	kien	supergroup	20.36 MB	Jan 04 11:33	2	128 MB	datapart2.csv	
<input type="checkbox"/>	-rw-r--r--	kien	supergroup	199.16 MB	Jan 04 11:34	2	128 MB	preprocesscsv.csv	

Showing 1 to 6 of 6 entries

Previous **1** Next

CHƯƠNG 5. Xử lý dữ liệu và visualize

5.1. Chuẩn bị:

- Cài đặt Spark trên 1 cụm máy ảo và cài zeppelin trên máy master để xử lý.

URL: spark://192.168.3.11:7077

Alive Workers: 2

Cores in use: 4 Total, 0 Used

Memory in use: 2.0 GiB Total, 0.0 B Used

Resources in use:

Applications: 0 Running, 0 Completed

Drivers: 0 Running, 0 Completed

Status: ALIVE

Workers (2)

Worker Id	Address	State	Cores	Memory	Resources
worker-20220104201650-192.168.3.13-38965	192.168.3.13:38965	ALIVE	2 (0 Used)	1024.0 MiB (0.0 B Used)	
worker-20220104201753-192.168.3.12-39777	192.168.3.12:39777	ALIVE	2 (0 Used)	1024.0 MiB (0.0 B Used)	

Zeppelin Notebook Job anonymous

Welcome to Zeppelin!

Zeppelin is web-based notebook that enables interactive data analytics.
You can make beautiful data-driven, interactive, collaborative document with SQL, code and even more!

Notebook

- Import note
- Create new note

- Flink Tutorial
- Miscellaneous Tutorial
- Python Tutorial
- R Tutorial
- Spark Tutorial
- LoveBigdata
- MLspark

Help

Get started with [Zeppelin documentation](#)

Community

Please feel free to help us to improve Zeppelin,
Any contribution are welcome!

- Mailing list
- Issues tracking
- Github

•

5.2. Xử lý dữ liệu từ HDFS:

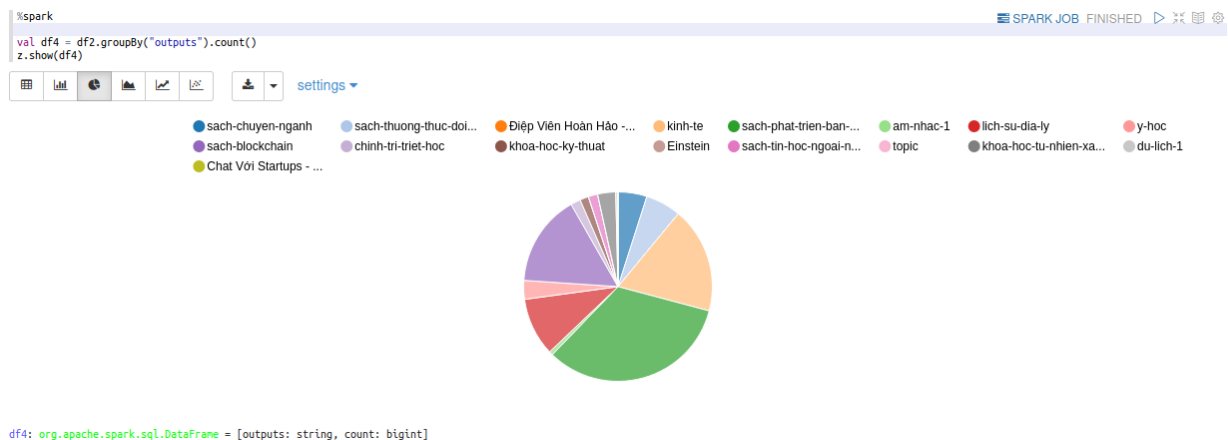
- Ta dùng spark kết hợp với các câu lệnh trong zeppelin đọc dữ liệu từ HDFS:

```
%spark
val df2 = spark.read.option("parserLib", "univocity").option("header", true)
    .option("multiline", true).options(Map("delimiter" -> ";"))csv("hdfs://nodens:9000/user/kien/BTL/data3.csv")
df2.printSchema
df2.show()
```

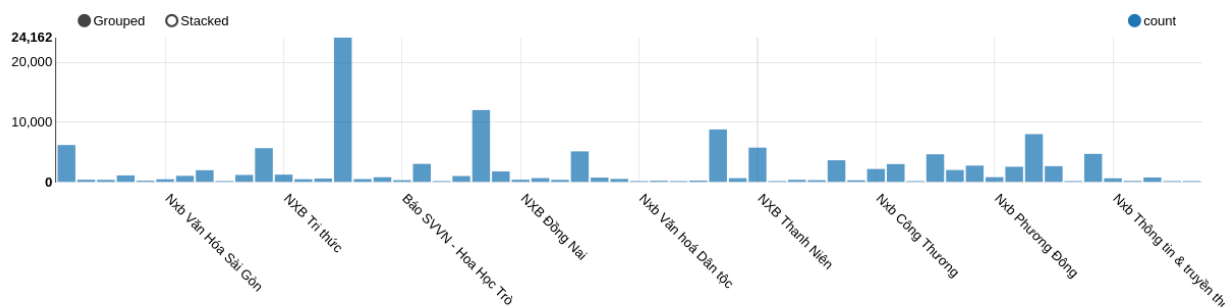
STT	topic	introduction	name	price	cover	authors	translator	pub
lisher	printer	id	weight	language	format	width	height	publish_date
0	sach-blockchain	"Một năm sau khi ... trải nghiệm nhiề... thích mê những d... nói hộ tâm tình ... bạn sẽ thấy lòng... nhớ tiếc cũng nh... phận người và ng... tất cả chúng ta có đi nhiều nơi mê mãi đủ chốn t... vẫn là trong-lớn... Đường hai ngã ng... Ngày trời về phi... Hội viên Hội nhà... bỏ qua những điều... "" Đường hai ngã ...						
1	sach-blockchain	"Chuyện có con nh... những khao khát ... thiết yếu và đượ... như người phụ nữ... thói quen hay cá... đàn bà cần một l... đàn ông cần một ... thấy cũng vẫn đẹ... nhưng không kén ... có thêm rất nhiề... trong vỏ thức ch... thềm được gặp gỡ thềm được một vò... một bờ môi nóng ... những nghi suy phát hiện về đời						
2	sach-blockchain	Một ngày u buồn, ... Lệnh Đệnh Tuổi 20		67000	https://www.vinab...		Ngô Thuận	null NXB Văn hóa - 16-Jun
Văn...	Phương Nam	1.13E+12	198	Tiếng Việt	Bìa nền	0	20	

5.2. Visualize dữ liệu thu thập được trên zeppelin:

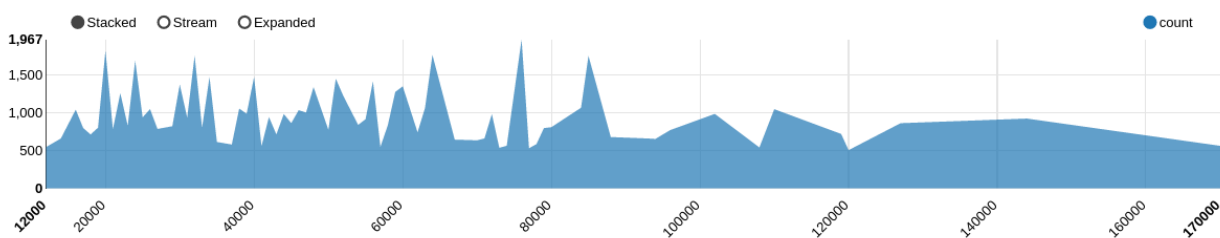
a. Tổng hợp các loại sách có trên trang web:



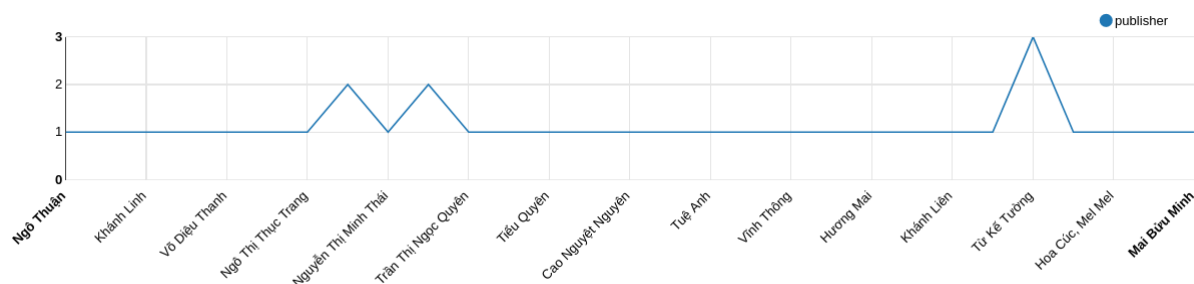
b. Số sách của nhà xuất bản có trên trang web:



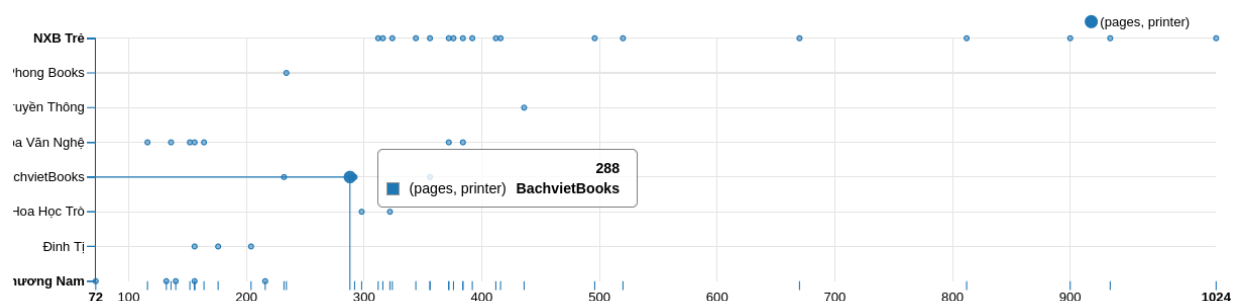
c. Biểu đồ về giá bán các loại sách trên trang web:



d. Số sách mà các tác giả thuộc “nhà xuất bản Văn hóa văn nghệ” đã đưa ra thi trường:



e. Các người xuất bản đã in bao nhiêu trang sách:



- Khi chạy trên zeppelin thì các job sẽ được thực thi liên tục cho đến khi nào chương trình kết thúc:

▼ Completed Jobs (107)

Page: 1 2 >

2 Pages. Jump to 1 . Show 100 items in a page. Go

Job Id (Job Group) ▼	Description	Submitted	Duration	Stages: Succeeded/Total	Tasks (for all stages): Succeeded/Total
106 (zeppelin anonymous 2GSAR67Q9 paragraph_1641227887111_674859770)	Started by: anonymous takeAsList at Spark3Shims.java:74	2022/01/04 13:42:48	6 s	1/1	1/1
105 (zeppelin anonymous 2GSAR67Q9 paragraph_1641227887111_674859770)	Started by: anonymous takeAsList at Spark3Shims.java:74	2022/01/04 13:32:03	2 s	1/1 (1 skipped)	75/75 (1 skipped)
104 (zeppelin anonymous 2GSAR67Q9 paragraph_1641227887111_674859770)	Started by: anonymous takeAsList at Spark3Shims.java:74	2022/01/04 13:32:01	3 s	1/1 (1 skipped)	100/100 (1 skipped)
103 (zeppelin anonymous 2GSAR67Q9 paragraph_1641227887111_674859770)	Started by: anonymous takeAsList at Spark3Shims.java:74	2022/01/04 13:32:00	0.9 s	1/1 (1 skipped)	20/20 (1 skipped)
102 (zeppelin anonymous 2GSAR67Q9 paragraph_1641227887111_674859770)	Started by: anonymous takeAsList at	2022/01/04 13:32:00	0.2 s	1/1 (1 skipped)	4/4 (1 skipped)

CHƯƠNG 6. Khó khăn và hướng phát triển

- **Khó khăn:** Do tình hình dịch bệnh covid nên cả nhóm không thể trực tiếp gặp mặt để triển khai công việc nên mọi việc cài đặt chỉ do 1 bạn đảm nhiệm. Do cấu hình máy kém nên việc triển khai mất rất nhiều thời gian và khó khăn.
- **Hướng phát triển:** Do thời gian xử lý và làm sạch nguồn dữ liệu lớn mất khá nhiều thời gian nên kết quả của chúng em đạt được cho bài tập lớn vẫn chưa như ý muốn. Nhóm em cũng đã chuẩn bị xong 1 file dữ liệu lớn để áp dụng mô hình học máy để phân loại các sách thu được. Nếu được chúng em sẽ cố gắng thử áp dụng học máy vào bigdata này.