

# BÁO CÁO TIỂU LUẬN CUỐI KÌ

Học phần: Khoa Học Dữ liệu

Giảng viên giảng dạy: TS. Ninh Khánh Duy

Thành viên:

1. Lê Hữu Hưng
2. Hoàng Nguyên Bách
3. Đinh Huy Hoàng

Lớp sinh hoạt: 20TCLC\_KHDL

Nhóm học phần: 20nh15

# Cấu trúc bài tiểu luận

- Giới thiệu đề tài
- Thu thập và mô tả dữ liệu
- Trích xuất đặc trưng
- Mô hình hóa dữ liệu
- Kết luận
- Tài liệu tham khảo



# Giới thiệu về đề tài

## Mục tiêu

Xây dựng mô hình phân loại thể loại của các bài báo.

- Đầu vào là của một bài báo bất kì.
- Đầu ra trả về là chủ đề chính của bài báo(bài báo đó thuộc lĩnh vực nào).

## Giải pháp

- Phương pháp giải quyết của chúng tôi là sử dụng PhoBert, một mô hình nhúng từ vựng tiếng Việt dựa trên kiến trúc Transformer

# Thu thập và mô tả dữ liệu



**Highlights | ĐT nữ Việt Nam 4-0 ĐT nữ Campuchia |  
Bán kết bóng đá nữ SEA Games 32**

SEA Games 32 - 12/05/2023

VTVCB - Tung ra sân đội hình có tới 8 sự thay đổi so với trận đấu trước, tuy nhiên ĐT nữ Việt Nam vẫn tỏ ra quá mạnh so với đối thủ và giành chiến thắng đậm.

Mỗi bài báo trên website sau khi thu thập về sẽ gồm có những nội dung như sau:

- Tiêu đề.
- Chủ đề.

# Thu thập và mô tả dữ liệu

Từ những dữ liệu trên nhóm đã thu thập thành công được hai tập dữ liệu.

- Small data: gồm 1374 mẫu.
- Big data: gồm 10981 mẫu.

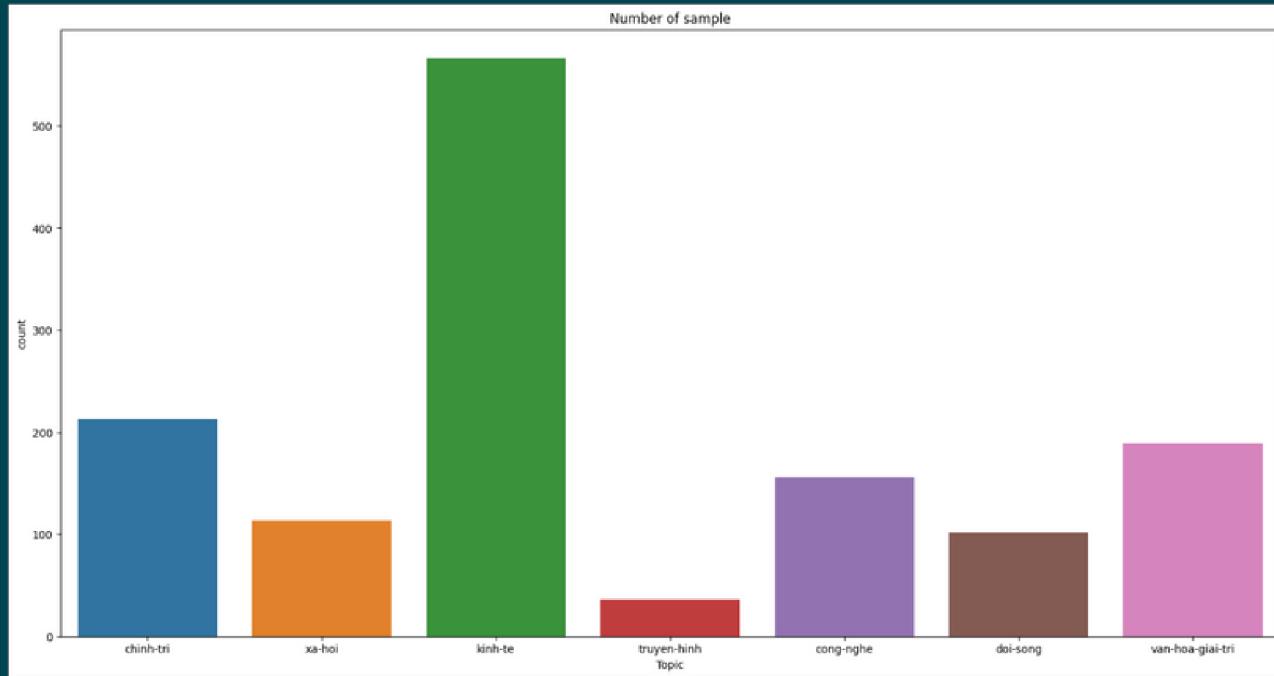
Small Data

Tiêu đề	Số lượng
kinh-te	566
truyen-hinh	213
chinh-tri	189
van-hoa-giai-tri	156
doi-song	113
xa-hoi	102
cong-nghe	36

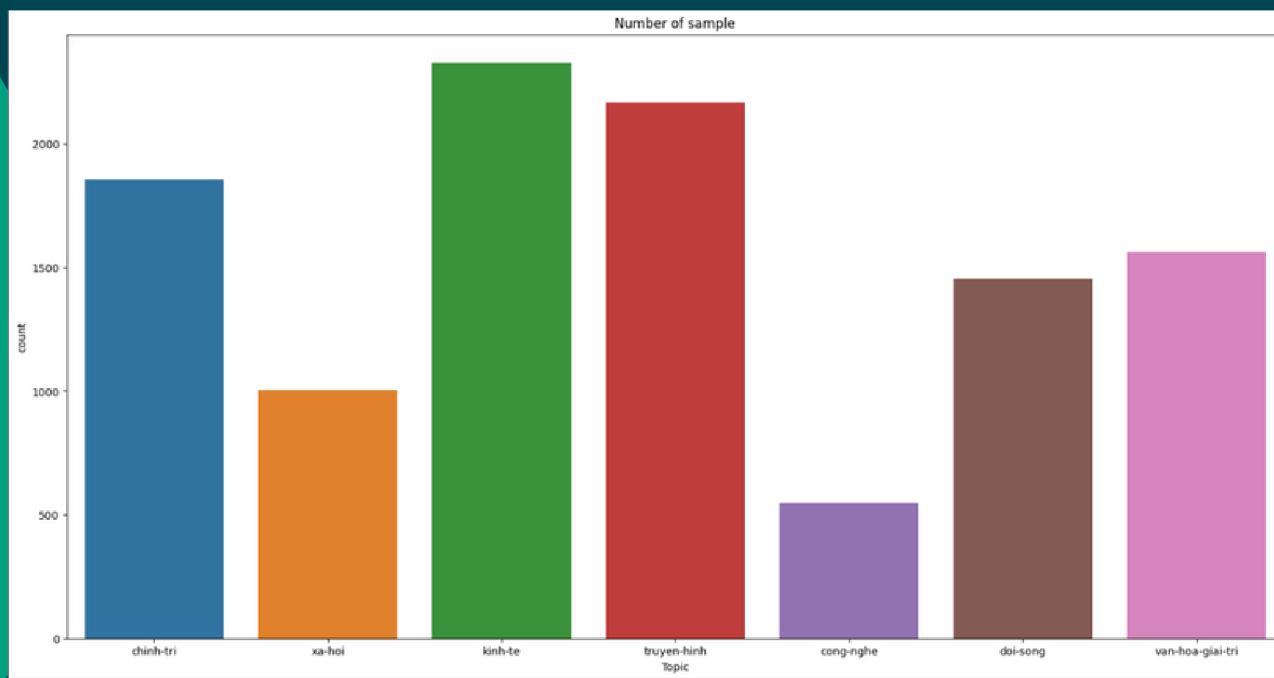
Big Data

Tiêu đề	Số lượng
kinh-te	2326
truyen-hinh	2164
chinh-tri	1856
van-hoa-giai-tri	1564
doi-song	1456
xa-hoi	1005
cong-nghe	548

# Số lượng topic trên mỗi tập dữ liệu (trước khi xử lý)

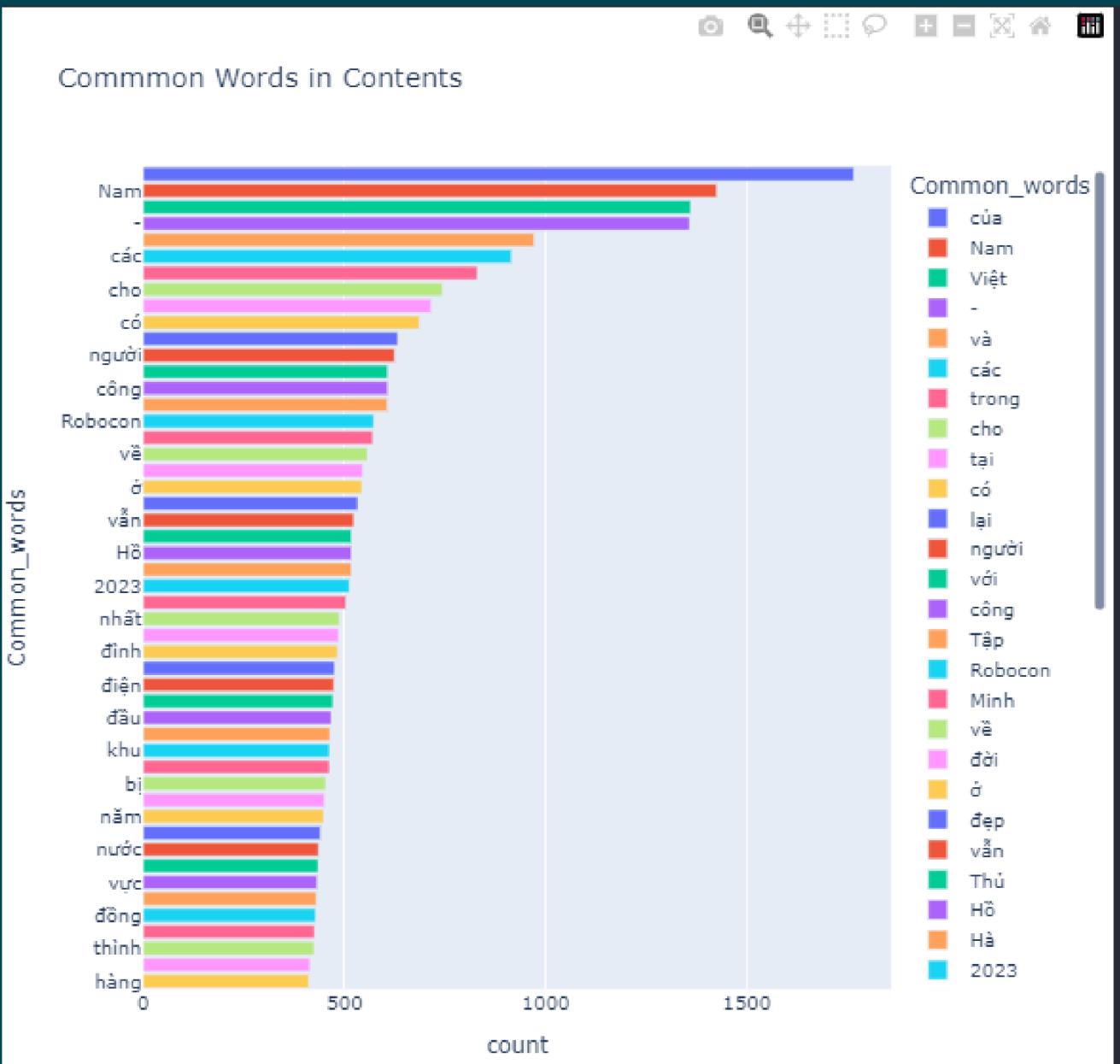


Small data

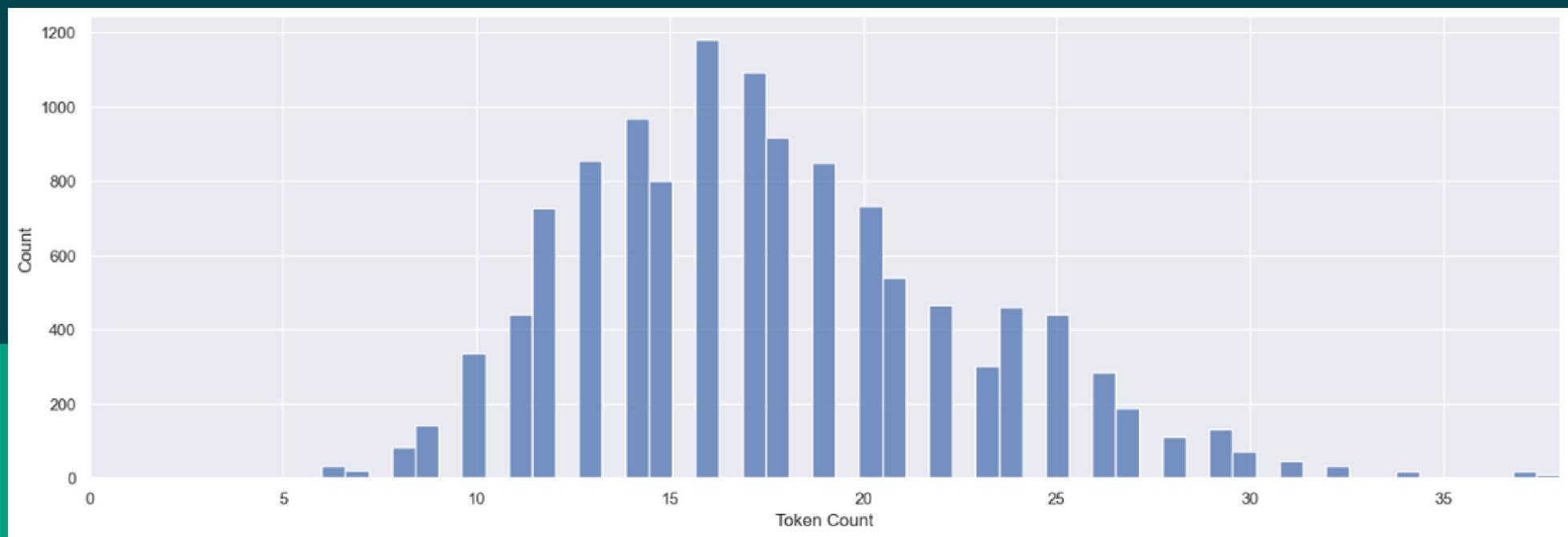


Big data

# Tần suất xuất hiện của các từ trong contents (trước khi xử lý)



# Số lượng từ trong mỗi câu (trước xử lý)

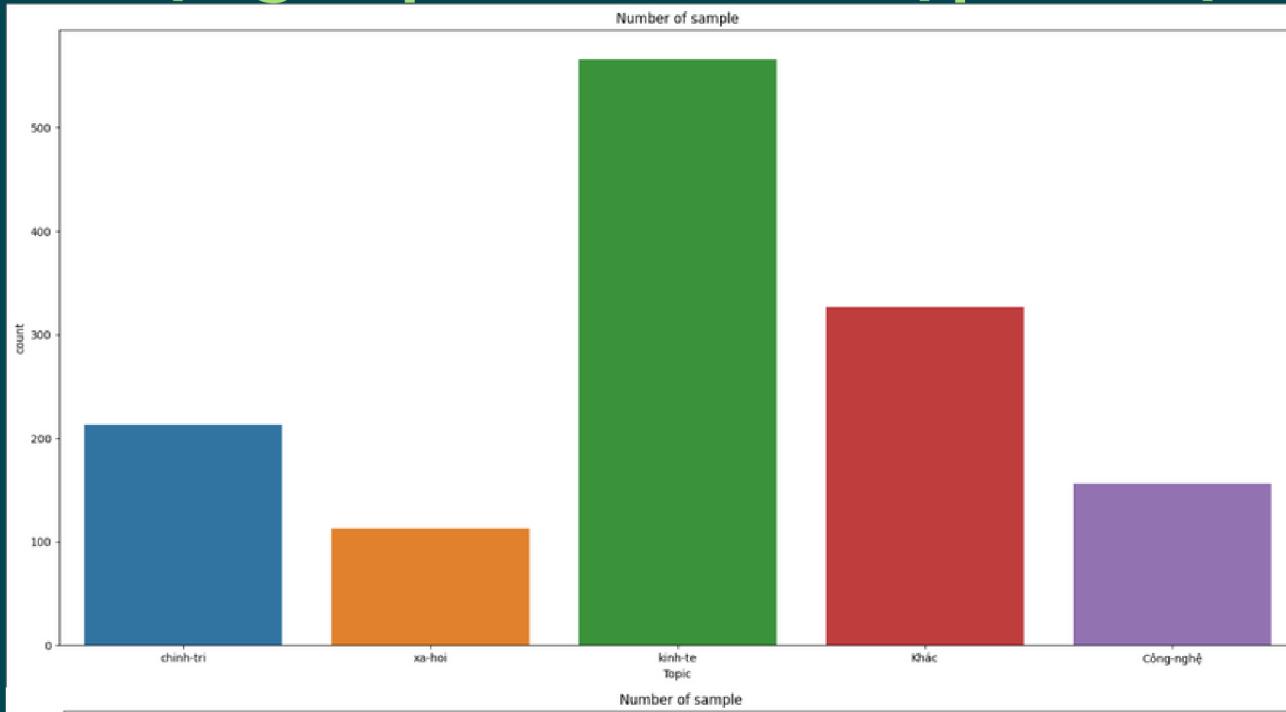


# Tần suất xuất hiện các từ trong contents (trước khi xử lý)

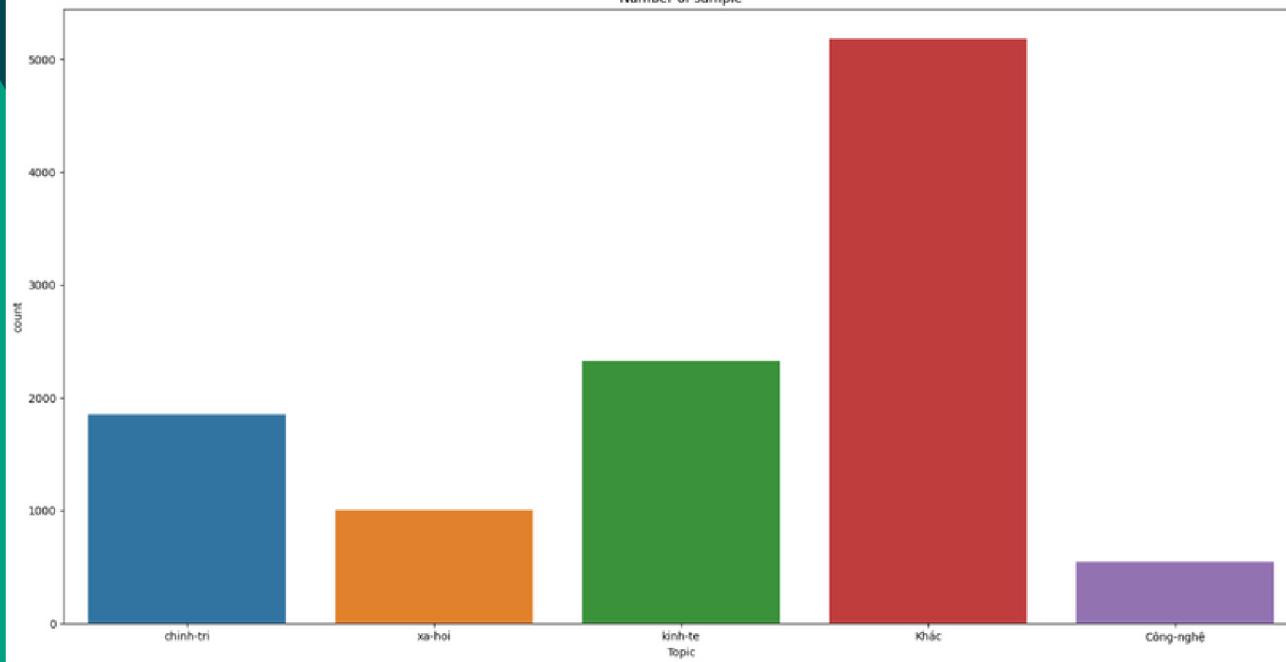
The word cloud displays the following words and their associated descriptive labels:

- trong
- vẫn
- tại
- Có
- điện
- không
- nước
- động
- Quốc
- vui
- vòng
- định
- mình
- lại
- đẹp
- nhất
- Hà
- năm
- với
- Minh
- sao
- Viet
- công
- đồng
- đầu
- về
- thình
- Cuộc
- bị
- nhất
- Thủ
- được
- Tập
- Cá
- vực
- đời
- người
- khu
- Hồ
- bất
- hang
- Robocon
- Thủ
- được
- Cá
- cho
- hội

# Số lượng topic trên mỗi tập dữ liệu (sau khi xử lý)

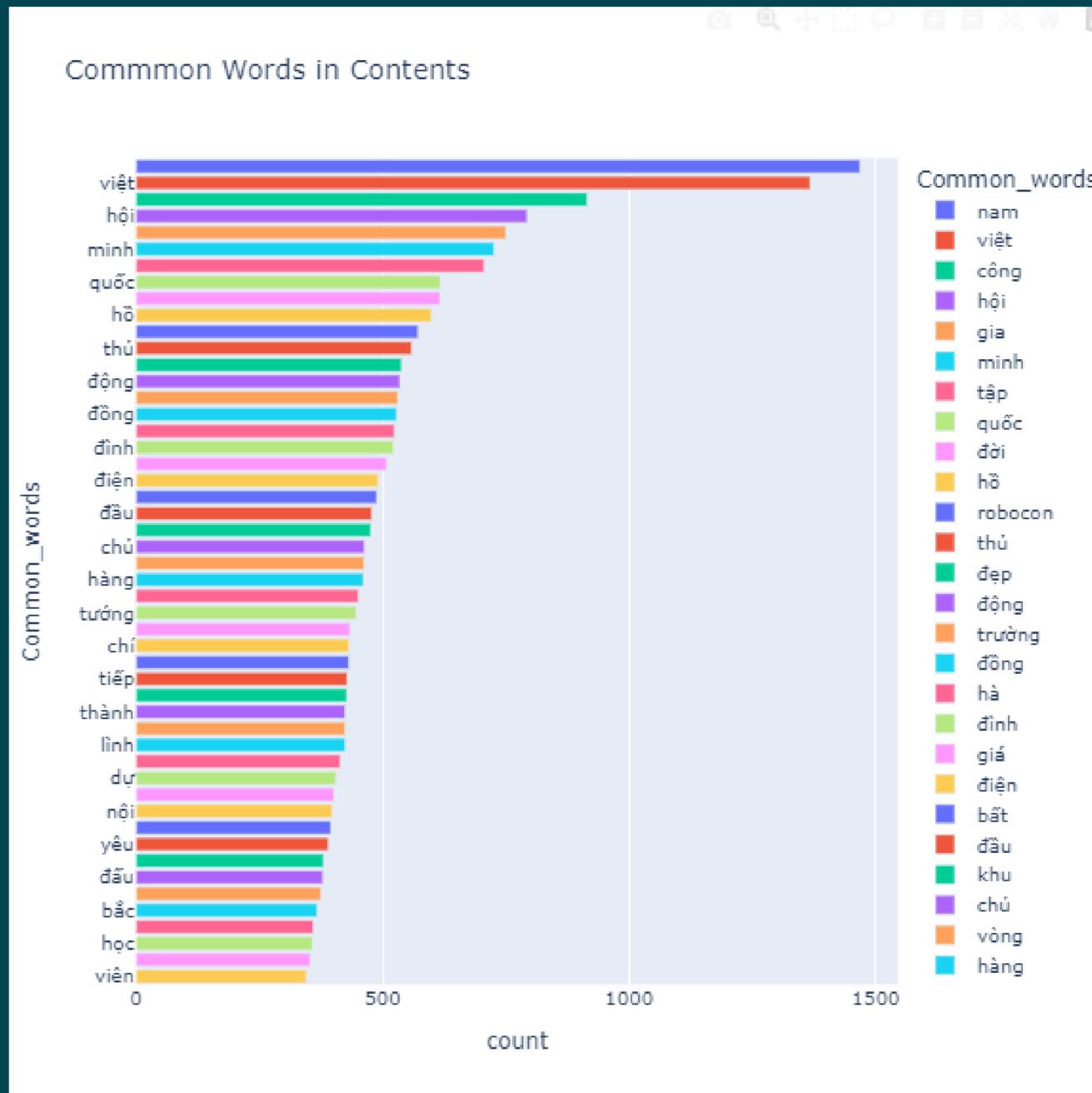


Small data



Big data

# Tần suất xuất hiện các từ trong contents (sau khi xử lý)



# Số lượng từ trong mỗi câu (sau xử lý)



# Tần suất xuất hiện các từ trong contents (sau khi xử lý)

## Nhận xét

- Sau khi dữ liệu thì số lượng nhãn giảm xuống chỉ còn 5 nhãn.
- Số lượng các kí hiệu đặc biệt dấu câu giảm đi đáng kể.
- Số lượng từ dừng giảm đi nhiều giúp cho các câu hầu như điều mang từ có nghĩa.
- Số lượng từ trong mỗi câu trước khi xử lý có giá trị max là 40 từ, sau khi xử lý chỉ còn 30 từ.
- Chuyển từ hoa thành từ thường giúp cho tokenizer hiệu quả hơn.

# Trích xuất đặc trưng



## Chọn đặc trưng

- + Xóa các từ dừng
  - + Xóa các ký tự đặc biệt
  - + Xóa các chữ số
- 

## Tokenizer

Chuyển câu thành các token được lấy trong từ điển.

---

## Embedding

Chuyển các câu sau khi đã được tokenizer về vecto embedding.

# Mô hình

```
RobertaModel(  
    (embeddings): RobertaEmbeddings(  
        (word_embeddings): Embedding(64001, 768, padding_idx=1)  
        (position_embeddings): Embedding(258, 768, padding_idx=1)  
        (token_type_embeddings): Embedding(1, 768)  
        (LayerNorm): LayerNorm((768,), eps=1e-05, elementwise_affine=True)  
        (dropout): Dropout(p=0.1, inplace=False)  
    )  
    (encoder): RobertaEncoder(  
        (layer): ModuleList(  
            (0): RobertaLayer(  
                (attention): RobertaAttention(  
                    (self): RobertaSelfAttention(  
                        (query): Linear(in_features=768, out_features=768, bias=True)  
                        (key): Linear(in_features=768, out_features=768, bias=True)  
                        (value): Linear(in_features=768, out_features=768, bias=True)  
                        (dropout): Dropout(p=0.1, inplace=False)  
                    )  
                    (output): RobertaSelfOutput(  
                        (dense): Linear(in_features=768, out_features=768, bias=True)  
                        (LayerNorm): LayerNorm((768,), eps=1e-05, elementwise_affine=True)  
                        (dropout): Dropout(p=0.1, inplace=False)  
                    )  
                )  
                (intermediate): RobertaIntermediate(  
...  
                    (pooler): RobertaPooler(  
                        (dense): Linear(in_features=768, out_features=768, bias=True)  
                        (activation): Tanh()  
                    )  
                )  
            )  
        )  
    )
```

## Câu đầu vào

```
"Tôi là sinh viên đại học Khoa học Tự nhiên"
```

```
['Tôi', 'là', 'sinh', 'viên', 'đại', 'học', 'Khoa', 'học', 'Tự', 'nhiên']
```

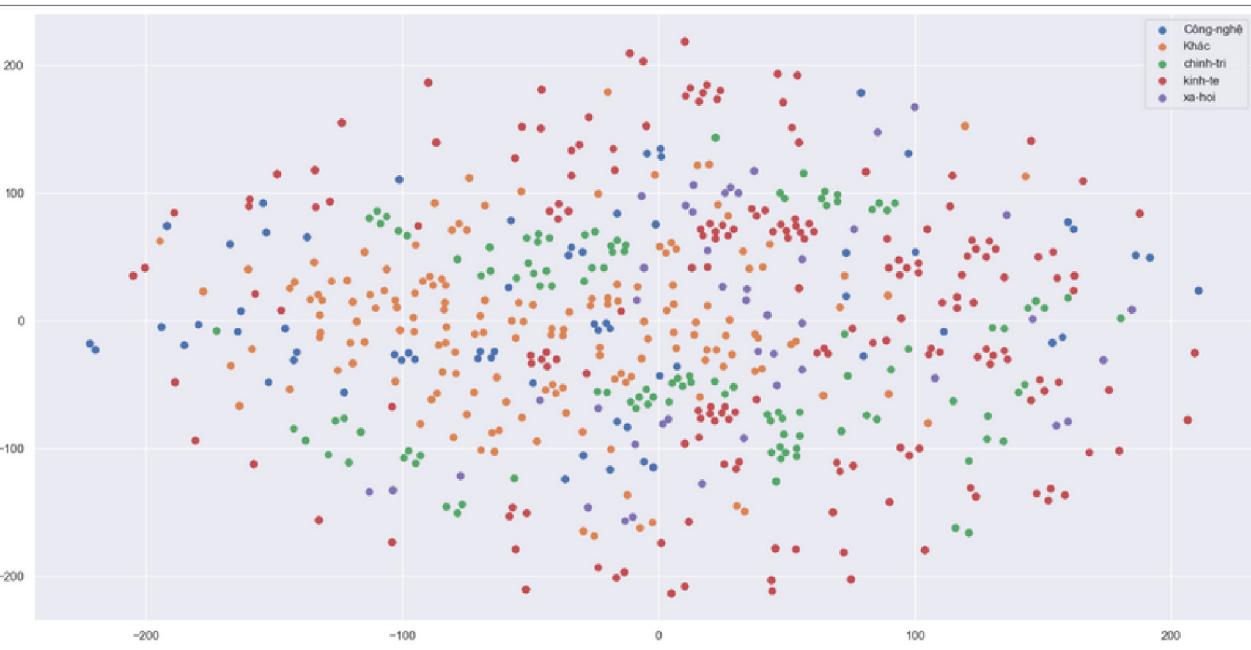
## Tokenizer

```
[218, 8, 418, 1430, 2919, 222, 3720, 222, 4544, 8249]
```

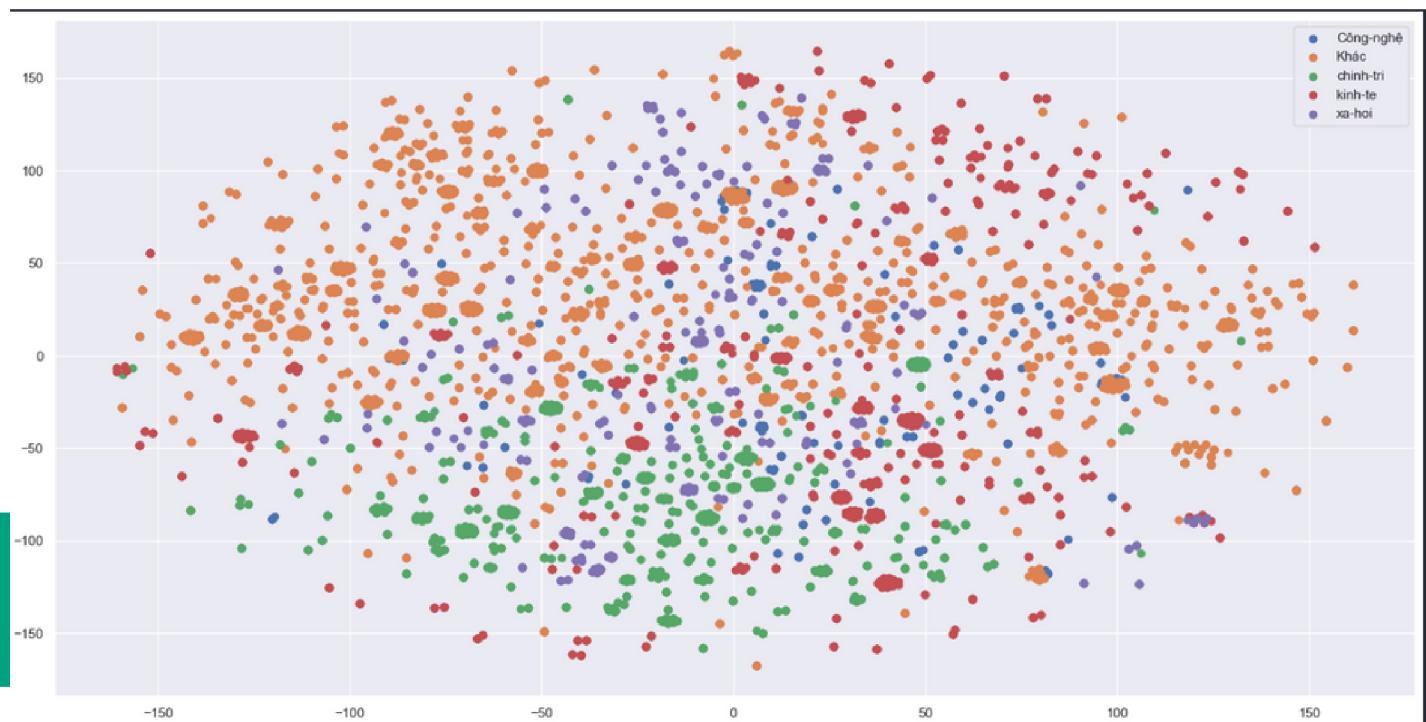
## Embedding

Sau khi đã Tokenizer, câu có độ dài khác nhau thì độ dài của Tokenizer sẽ khác nhau.

Vì thế, các câu sau khi đã được tokenizer sẽ được đi qua pre-trained model PhoBert để tiến hành embedding về cùng số chiều 768.

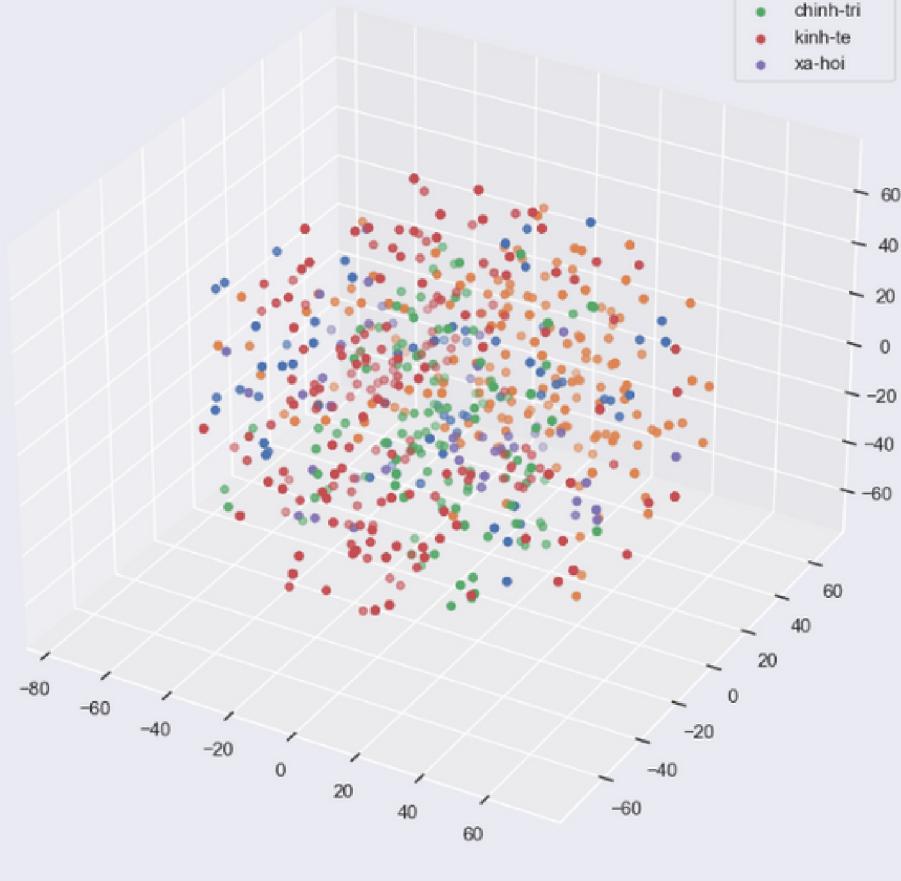


Small data



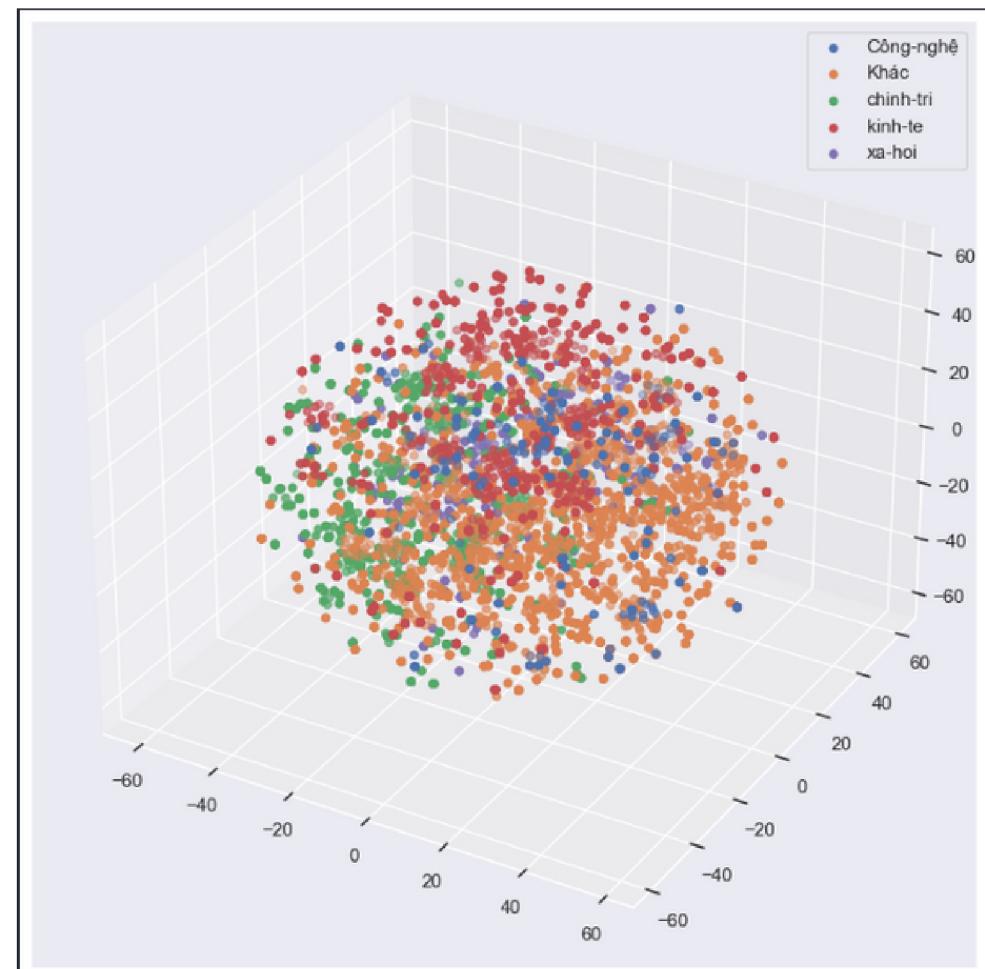
Big data

- Công-nghệ
- Khác
- chính-trị
- kinh-te
- xa-hoi



Big data

Small data



## Nhận Xét:

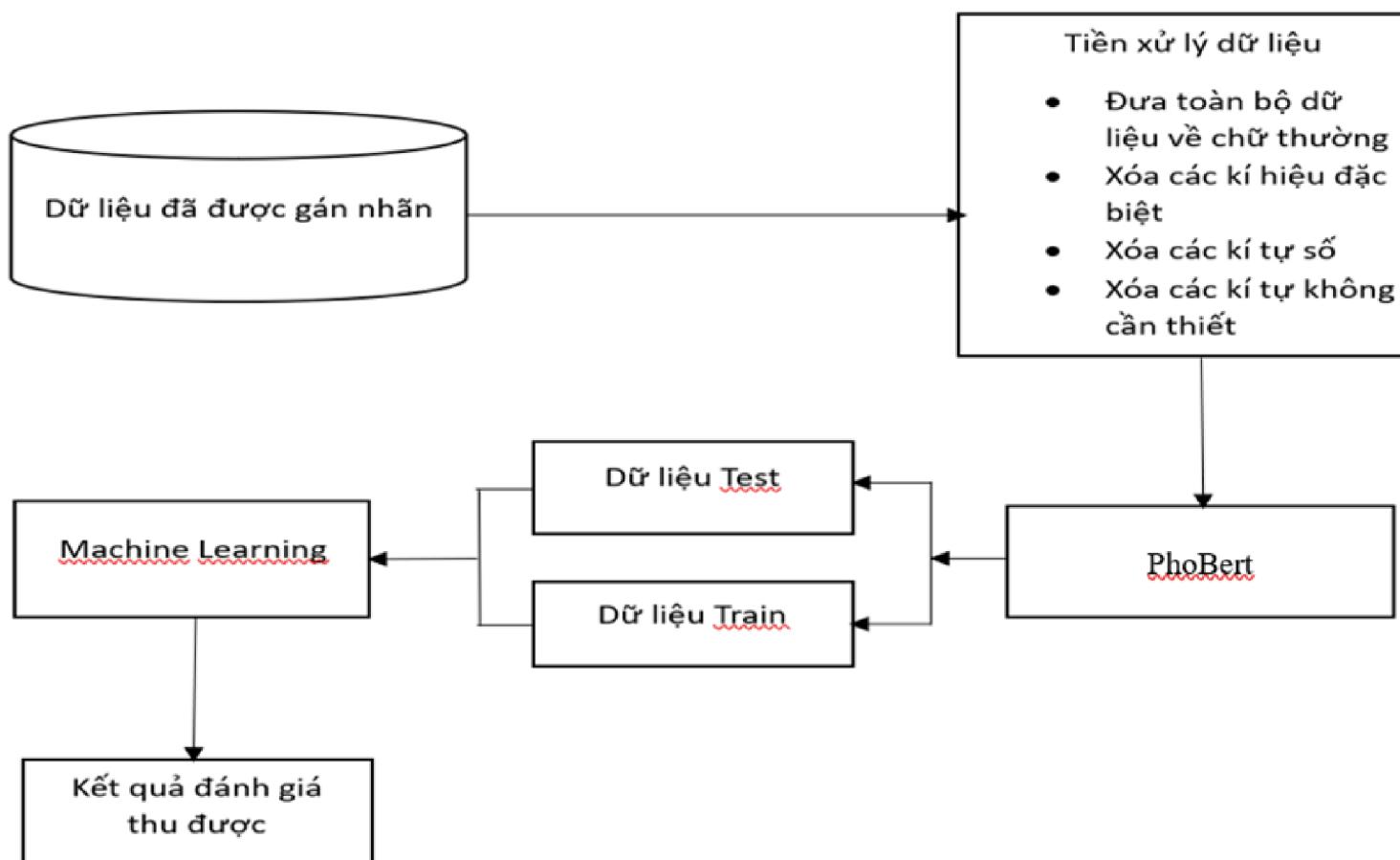
- Khi trực quan hóa vector đặc trưng 768 chiều của mỗi contents về vector 3 chiều ta thấy tập dữ liệu lớn phân cụm rõ ràng hơn tập dữ liệu nhỏ.
- 2d cx như vậy tập dữ liệu lớn cho ta thấy sự phân cụm rõ ràng hơn
- Điều này cho ta thấy các vector có cùng 1 topic sẽ có xu hướng ở gần nhau hơn trong không gian vector.

# Mô hình hóa dữ liệu

## Phân chia dữ liệu

- Repeated hold-out
- Cross-validation

[Quay lại Trang Chương trình](#)



# Thuật toán

- Support Vector Machine (SVM)
- Random Forest Classifier

# Đánh giá thuật toán

- Accuracy
- Recall
- F1 Score
- Ma trận nhầm lẫn

# Kết luận

- Small data

	Dataset	Normalize	Dimension	Model	evaluation	accuracy	recall	f1
0	DataFrame	None	128	SVC	training_model_repeat_holdout	0.932446	0.913474	0.918216
1	DataFrame	None	128	SVC	training_model_cross_val	0.941818	0.921828	0.924988
2	DataFrame	None	128	RandomForestClassifier	training_model_repeat_holdout	0.929540	0.898803	0.915835
3	DataFrame	None	128	RandomForestClassifier	training_model_cross_val	0.947636	0.923873	0.938936
4	DataFrame	None	256	SVC	training_model_repeat_holdout	0.936804	0.916962	0.923382
5	DataFrame	None	256	SVC	training_model_cross_val	0.944000	0.923152	0.926640
6	DataFrame	None	256	RandomForestClassifier	training_model_repeat_holdout	0.922518	0.894282	0.909681
7	DataFrame	None	256	RandomForestClassifier	training_model_cross_val	0.945455	0.922584	0.936908
8	DataFrame	None	512	SVC	training_model_repeat_holdout	0.938257	0.918937	0.925011
9	DataFrame	None	512	SVC	training_model_cross_val	0.944727	0.924442	0.928273
10	DataFrame	None	512	RandomForestClassifier	training_model_repeat_holdout	0.921308	0.893304	0.908483
11	DataFrame	None	512	RandomForestClassifier	training_model_cross_val	0.952000	0.927861	0.941752
12	DataFrame	None	768	SVC	training_model_repeat_holdout	0.938257	0.918937	0.925011
13	DataFrame	None	768	SVC	training_model_cross_val	0.944727	0.924442	0.928273
14	DataFrame	None	768	RandomForestClassifier	training_model_repeat_holdout	0.935109	0.904388	0.922613
15	DataFrame	None	768	RandomForestClassifier	training_model_cross_val	0.952000	0.926520	0.943280
16	DataFrame	Norm	128	SVC	training_model_repeat_holdout	0.811138	0.719598	0.750605
17	DataFrame	Norm	128	SVC	training_model_cross_val	0.830545	0.753908	0.785808
18	DataFrame	Norm	128	RandomForestClassifier	training_model_repeat_holdout	0.924213	0.895064	0.912620
19	DataFrame	Norm	128	RandomForestClassifier	training_model_cross_val	0.941818	0.919397	0.935462
20	DataFrame	Norm	256	SVC	training_model_repeat_holdout	0.842857	0.763052	0.794309
21	DataFrame	Norm	256	SVC	training_model_cross_val	0.864000	0.799051	0.827717
22	DataFrame	Norm	256	RandomForestClassifier	training_model_repeat_holdout	0.921308	0.893477	0.909015
23	DataFrame	Norm	256	RandomForestClassifier	training_model_cross_val	0.944000	0.921092	0.935504
24	DataFrame	Norm	512	SVC	training_model_repeat_holdout	0.844552	0.767694	0.798750
25	DataFrame	Norm	512	SVC	training_model_cross_val	0.864727	0.799657	0.829067
26	DataFrame	Norm	512	RandomForestClassifier	training_model_repeat_holdout	0.921792	0.894188	0.909672
27	DataFrame	Norm	512	RandomForestClassifier	training_model_cross_val	0.952727	0.929075	0.943232
28	DataFrame	Norm	768	SVC	training_model_repeat_holdout	0.844552	0.767694	0.798750
29	DataFrame	Norm	768	SVC	training_model_cross_val	0.864727	0.799657	0.829067
30	DataFrame	Norm	768	RandomForestClassifier	training_model_repeat_holdout	0.929782	0.899610	0.917705
31	DataFrame	Norm	768	RandomForestClassifier	training_model_cross_val	0.947636	0.923687	0.939737

# Kết luận

- Big data

	Dataset	Normalize	Dimension	Model	evaluation	accuracy	recall	f1
0	DataFrame	None	128	SVC	training_model_repeat_holdout	0.932688	0.913221	0.918423
1	DataFrame	None	128	SVC	training_model_cross_val	0.941818	0.921567	0.925894
2	DataFrame	None	128	RandomForestClassifier	training_model_repeat_holdout	0.931477	0.902062	0.918583
3	DataFrame	None	128	RandomForestClassifier	training_model_cross_val	0.944000	0.920720	0.936579
4	DataFrame	None	256	SVC	training_model_repeat_holdout	0.937046	0.917252	0.923379
5	DataFrame	None	256	SVC	training_model_cross_val	0.944727	0.924442	0.927438
6	DataFrame	None	256	RandomForestClassifier	training_model_repeat_holdout	0.922276	0.894554	0.910036
7	DataFrame	None	256	RandomForestClassifier	training_model_cross_val	0.950545	0.927304	0.940755
8	DataFrame	None	512	SVC	training_model_repeat_holdout	0.938257	0.918937	0.925011
9	DataFrame	None	512	SVC	training_model_cross_val	0.944727	0.924442	0.928273
10	DataFrame	None	512	RandomForestClassifier	training_model_repeat_holdout	0.921308	0.893304	0.908483
11	DataFrame	None	512	RandomForestClassifier	training_model_cross_val	0.952000	0.927861	0.941752
12	DataFrame	None	768	SVC	training_model_repeat_holdout	0.938257	0.918937	0.925011
13	DataFrame	None	768	SVC	training_model_cross_val	0.944727	0.924442	0.928273
14	DataFrame	None	768	RandomForestClassifier	training_model_repeat_holdout	0.935109	0.904388	0.922613
15	DataFrame	None	768	RandomForestClassifier	training_model_cross_val	0.952000	0.926520	0.943280
16	DataFrame	Norm	128	SVC	training_model_repeat_holdout	0.815738	0.723587	0.756102
17	DataFrame	Norm	128	SVC	training_model_cross_val	0.834182	0.755735	0.787360
18	DataFrame	Norm	128	RandomForestClassifier	training_model_repeat_holdout	0.925182	0.894831	0.912228
19	DataFrame	Norm	128	RandomForestClassifier	training_model_cross_val	0.944727	0.921336	0.937553
20	DataFrame	Norm	256	SVC	training_model_repeat_holdout	0.842131	0.762001	0.793301
21	DataFrame	Norm	256	SVC	training_model_cross_val	0.865455	0.800907	0.830001
22	DataFrame	Norm	256	RandomForestClassifier	training_model_repeat_holdout	0.922518	0.894714	0.910048
23	DataFrame	Norm	256	RandomForestClassifier	training_model_cross_val	0.946182	0.923537	0.937430
24	DataFrame	Norm	512	SVC	training_model_repeat_holdout	0.844552	0.767694	0.798750
25	DataFrame	Norm	512	SVC	training_model_cross_val	0.864727	0.799657	0.829067
26	DataFrame	Norm	512	RandomForestClassifier	training_model_repeat_holdout	0.921792	0.894188	0.909672
27	DataFrame	Norm	512	RandomForestClassifier	training_model_cross_val	0.952727	0.929075	0.943232
28	DataFrame	Norm	768	SVC	training_model_repeat_holdout	0.844552	0.767694	0.798750
29	DataFrame	Norm	768	SVC	training_model_cross_val	0.864727	0.799657	0.829067
30	DataFrame	Norm	768	RandomForestClassifier	training_model_repeat_holdout	0.929782	0.899610	0.917705
31	DataFrame	Norm	768	RandomForestClassifier	training_model_cross_val	0.947636	0.923687	0.939737

# Kết luận

## Trên cùng một tập dữ liệu

- Chia tập dữ liệu để huấn luyện mô hình bằng 2 phương pháp: repeat hold-out và cross-validate thì các metric đánh giá độ chính xác cho ra kết quả chênh lệch không lớn (1-2%). Nhưng trong các trường hợp cross-validate luôn nhỉnh hơn về độ chính xác so với repeat hold-out.
- Sau khi trích xuất đặc trưng, chuẩn hóa các vector đặc trưng: thì các metric đánh giá độ chính xác giảm đi khá nhiều(5- 7%). Nguyên nhân là vì sử dụng mô hình trích xuất đặc trưng pre-train đã quá tốt, nên khi chuẩn hóa sẽ làm giảm độ chính xác của các metric đánh giá đo được.

# Kết luận

Trên cùng một tập dữ liệu

Đối với giảm chiều dữ liệu:

- 512(dimension): 95.20% (RandomForestClassifier - cross-validate)
- 256(dimension): 94.54% (RandomForestClassifier - cross-validate)
- 128(dimension): 94.76% (RandomForestClassifier - cross-validate)
- So với 768(dimension) không giảm chiều: 95.20% (RandomForestClassifier - cross-validate).

Vậy giảm chiều dữ liệu mà không làm giảm đi độ chính xác của mô hình, khuyến khích sử dụng giảm chiều dữ liệu để cho mô hình xử lý nhanh và hiệu quả hơn.

Tổng kết: So sánh được mức độ hiệu quả hay sự chênh lệch về độ chính xác của từng phương pháp với nhau dựa trên bảng tổng hợp phía trên.

# Kết luận

## Trên hai tập dữ liệu khác nhau

Thực hiện mô hình hóa dữ liệu với hai tập dữ liệu khác nhau là small data (~1000 mẫu) và big data (~10000 mẫu) => Độ chính xác khi áp dụng các phương pháp giống nhau cho cả hai tập thì độ chính xác chênh lệch không lớn. Chênh lệch từ (2-3%) đối với mỗi phương pháp.

Tập dữ liệu lớn (big data) luôn luôn có độ chính xác cao hơn, vì mô hình được huấn luyện nhiều hơn nên có tính khái quát hóa cao hơn.

Thời gian huấn luyện của tập dữ liệu lớn sẽ nhiều hơn so với tập dữ liệu nhỏ.

Độ chính xác của mô hình sẽ đáng tin cậy hơn trên tập dữ liệu lớn so với tập dữ liệu nhỏ.

# Kết luận

## Giới hạn của đề tài

- Đề tài đã hoàn thành các yêu cầu, mục tiêu đặt ra. Tuy nhiên tập dữ liệu ban đầu chỉ ở mức thử nghiệm.
- Điểm hạn chế: Đề tài đang sử dụng mô hình Embedding chưa tinh chỉnh để phù hợp với yêu cầu đề tài.

# Tài liệu tham khảo

- BERT, RoBERTa, PhoBERT, BERTweet: Ứng dụng state-of-the-art pre-trained model cho bài toán phân loại văn bản ([viblo.asia](http://viblo.asia)).
- Nhận diện cảm xúc văn bản với PhoBERT, Hugging Face - Mì AI ([miae.vn](http://miae.vn)).

# THANKS FOR WATCHING

[Quay lại Trang Chương trình](#)