BỘ MÔN HỆ THỐNG THÔNG TIN – KHOA CÔNG NGHỆ THÔNG TIN

ĐẠI HỌC KHOA HỌC TỰ NHIÊN THÀNH PHỐ HỒ CHÍ MINH, ĐẠI HỌC QUỐC GIA TP HCM

MÔN HỌC THỐNG KÊ

Sinh viên thực hiện: 20120041 - Trần Kim Bảo

20120053 - Nguyễn Thành Đạt

20120056 - Trần Quốc Đỉnh

20120060 - Nguyễn Trí Đức

GV phụ trách: Ngô Minh Nhựt

ĐỒ ÁN MÔN HỌC - HỌC THỐNG KÊ

HỌC KỲ II – NĂM HỌC 2022-2023

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
227 Nguyễn Văn Cừ, Phường 4, Quận 5, TP.HCM
Điện Thoại: (08) 38.354.266 - Fax:(08) 38.350.096

BẢNG THÔNG TIN CHI TIẾT NHÓM

MSSV	Họ Tên	Email
20120041	Trần Kim Bảo	20120041@student.hcmus.edu.vn
20120053	Nguyễn Thành Đạt	20120053@student.hcmus.edu.vn
20120056	Trần Quốc Đỉnh	20120056@student.hcmus.edu.vn
20120060	Nguyễn Trí Đức	20120060@student.hcmus.edu.vn

MSSV	Công việc	Đánh giá
Trần Kim Bảo	Khám phá dữ liệu, Tìm hiểu và xây dựng mô hình.	100%
Nguyễn Thành Đạt	Tìm hiểu và xây dựng mô hình, tinh chỉnh siêu tham số mô hình, viết báo cáo.	100%
Trần Quốc Đỉnh	Tìm hiểu và xây dựng web app.	100%
Nguyễn Trí Đức	Tiền xử lý, tìm hiểu hỗ trợ xây dựng web app, viết báo cáo.	100%

Mục lục

A. Yêu cầu của Đồ án 3

B. Kết quả 3

1. Đề tài: Sentiment Analysis 3

2. Dữ liệu: Twitter Tweets Sentiment Analysis for Natural Language Processing 4

2.1 Cảm hứng lựa chọn 4

2.2 Nguồn gốc và giấy phép sử dụng 4

2.3 Thông tin chi tiết của dữ liệu 5

3. Mô hình: BERT 5

3.1 Tìm hiểu về mô hình BERT[3] 5

3.2 Quá trình huấn luyện 6

3.3 Đánh giá mô hình 7

4. Web app 8

4.3 Thư viện sử dụng: streamlit 8

4.4 Giới thiệu web[4]: 9

C. Tham khảo: 10

A. Yêu cầu của Đồ án

- Trước tiên, sinh viên sẽ tự chọn một bài toán trong lĩnh vực Xử lý Ngôn ngữ Tự nhiên để giải quyết. Gợi ý một số bài toán có thể chọn là:
  - Part of Speech (PoS) Tagging.
  - Text Classification.
  - Named Entity Recognition.
  - Machine Translation.
  - Text Summarization.
  - Question Answering.
  - ...
- Sau đó, sinh viên tiếp tục chọn một mô hình có nền tảng là kiến trúc Transformer để giải quyết bài toán được đặt ra và lần lượt thực hiện 2 yêu cầu sau:
  1. (7 điểm) Tìm một dataset phù hợp và huấn luyện lại mô hình trên bộ data đó. Sau đó, sinh viên cần thực hiện đánh giá mô hình bằng các độ đo phù hợp. Trong báo cáo, phải có mô tả chi tiết về tập dữ liệu, mô hình được sử dụng, cách phân chia dữ liệu (train/validation/test) và kết quả đánh giá mô hình bằng các độ đo phù hợp.
  2. (3 điểm) Xây dựng một ứng dụng có giao diện web từ mô hình vừa được huấn luyện.

B. Kết quả

1. Đề tài: Sentiment Analysis

- Sau khi tìm hiểu nhóm lựa chọn đề tài **Sentiment Analysis**, với các lý do sau:
  - Tính ứng dụng cao: Phân tích cảm xúc là một trong những lĩnh vực có tính ứng dụng cao nhất trong khoa học máy tính và trí tuệ nhân tạo. Nó có thể được áp dụng trong nhiều lĩnh vực khác nhau như thương mại điện tử, y tế, tài chính, và quản lý tri thức. Giúp các nhà đầu tư và các chuyên gia tài chính đánh giá được tình hình thị trường và dự đoán xu hướng giá cả. Điều này có thể giúp họ đưa ra quyết định đúng đắn về đầu tư và các hoạt động kinh doanh. Hay giúp các doanh nghiệp và tổ chức hiểu rõ

hơn về cách người dùng phản hồi đối với sản phẩm, dịch vụ hoặc sự kiện của họ. Điều này có thể giúp họ cải thiện sản phẩm hoặc dịch vụ của mình để đáp ứng nhu cầu của khách hàng.

+ Độ phức tạp tính toán thấp: Phân tích cảm xúc thường có độ phức tạp tính toán thấp hơn so với các lĩnh vực khác của trí tuệ nhân tạo như dự báo chuỗi thời gian hoặc mạng nơ-ron sâu. Phù hợp kiến thức của các thành viên trong nhóm.
+ Dữ liệu phong phú: Có rất nhiều loại dữ liệu khác nhau có thể được sử dụng để phân tích cảm xúc, từ dữ liệu văn bản đến hình ảnh và âm thanh. Điều này có nghĩa là phân tích cảm xúc có thể được áp dụng cho nhiều loại dữ liệu khác nhau.

2. Dữ liệu: Twitter Tweets Sentiment Analysis for Natural Language Processing

2.1 Cảm hứng lựa chọn

- Số lượng tweet lớn: Twitter là mạng xã hội lớn và phổ biến trên toàn cầu, với hàng triệu tweet được đăng tải hàng ngày. Điều này cung cấp một nguồn dữ liệu lớn để xây dựng mô hình phân loại nội dung.
- Tính đa dạng của nội dung: Trên Twitter, người dùng có thể chia sẻ mọi loại thông tin, từ tin tức, giải trí đến ý kiến cá nhân và cảm xúc. Do đó, tweet cung cấp một mẫu đa dạng của các loại nội dung phân loại cảm xúc.
- Tính ngắn gọn: Tweet có giới hạn ký tự 280, điều này yêu cầu người dùng phải tóm tắt ý kiến hoặc cảm xúc của họ trong một khoảng thời gian ngắn. Điều này làm cho tweet rất thú vị để phân tích, vì chúng ta cần đánh giá các từ và cụm từ để hiểu ý nghĩa của tweet.
- Tính thời sự: Twitter cung cấp một phương tiện để người dùng chia sẻ ý kiến và cảm xúc về các sự kiện thời sự đang diễn ra. Điều này cung cấp một cơ hội để xây dựng các mô hình phân loại nội dung về các sự kiện thời sự để theo dõi ý kiến của người dùng.

2.2 Nguồn gốc và giấy phép sử dụng

- Nhóm sử dụng dữ liệu: Twitter Tweets Sentiment Dataset[1]

- Giấy phép của dữ liệu: CC0(Public Domain)[2] - người dùng được phép sử dụng với mục đích phi thương mại và phải credit cho chủ sở hữu.
- Nguồn: Google với mục đích thu thập để nghiên cứu.

2.3 Thông tin chi tiết của dữ liệu

- Dữ liệu gồm: có 27841 dòng, 4 cột.
- Ý nghĩa của mỗi mẫu là chứa thông tin cơ bản của 1 tweet, bao gồm: mã định danh của tweet, nội dung, nội dung rút gọn và phân loại nội dung.
- Ý nghĩa của mỗi cột:

STT	Tên thuộc tính	Mô tả	Kiểu dữ liệu
1	textID	Mã định danh của bài viết	Object
2	text	Nội dung	Object
3	selected_text	Nội dung đã chắt lọc	Object
4	sentiment	Phân loại nội dung	Object

3. Mô hình: BERT

3.1 Tìm hiểu về mô hình BERT[3]

- BERT (Bidirectional Encoder Representations from Transformers): là mô hình ngôn ngữ tự nhiên dựa trên kiến trúc Transformer, được Google giới thiệu vào năm 2018. Mô hình này sử dụng phương pháp học chuyển tiếp (transformer) để học biểu diễn ngôn ngữ tự nhiên.

- BERT được huấn luyện trên một lượng lớn dữ liệu bằng cách đọc các đoạn văn bản ngôn ngữ tự nhiên từ các nguồn như Wikipedia và các tài liệu khác. Tuy nhiên, khác với các mô hình trước đó, BERT không chỉ đọc các đoạn văn bản từ trái sang phải, mà nó đọc đoạn văn bản theo cả hai hướng, từ trái sang phải và từ phải sang trái. Điều này giúp cho BERT hiểu được ngữ cảnh của từng từ trong câu và giúp cải thiện khả năng đưa ra dự đoán chính xác về ý nghĩa của câu.

- BERT sử dụng 2 phương pháp để huấn luyện mô hình:

- Masked language modeling (MLM): trong đó một số từ trong câu được che đi và mô hình phải dự đoán từ bị che. Quá trình này giúp mô hình học được cách biểu diễn từ vựng và ngữ pháp của ngôn ngữ tự nhiên một cách hiệu quả hơn.
- Next sentence prediction (NSP): trong đó mô hình được huấn luyện để dự đoán xem hai câu liên tiếp có phải là hai câu kế tiếp nhau trong văn bản hay không. Mục đích của phương pháp này là giúp cho BERT hiểu được mối quan hệ giữa các câu trong văn bản, giúp cho mô hình có khả năng xử lý các tác vụ như phân loại câu, trả lời câu hỏi và tóm tắt văn bản.

3.2 Quá trình huấn luyện

Các bước thực hiện:

- Bước 1: Label Encoding: Các nhãn được chuyển thành các giá trị số tương
- Bước 2: Khởi tạo mô hình pre-trained:
  - Khởi tạo mô hình đã được huấn luyện trên hugging face.
  - Các dữ liệu văn bản cần mã hoá về kiểu Tensor để thực hiện huấn luyện dữ liệu.
- Bước 3: Chia tập dữ liệu: Thực hiện chia tập dữ liệu thành 3 tập nhỏ hơn: train, test và validation bằng hàm train_test_split:
  - Tập train: 70%
  - Tập test: 15%
  - Tập validation: 15%

Bước 4: Huấn luyện mô hình:

- Thiết lập các tham số cho mô hình
  - ***batch_size = 64***: kích thước của mỗi lô dữ liệu để đưa vào mỗi lần khi train.
  - ***logging_steps = 100***: số lượng bước huấn luyện giữa mỗi lần ghi log.
  - ***output_dir = 'output'***: đường dẫn đến thư mục lưu trữ các file liên quan đến quá trình huấn luyện và đánh giá mô hình, bao gồm các checkpoint của mô hình, các file log và các file liên quan đến việc đánh giá mô hình.
  - ***num_train_epochs = 10***: số lần lặp lại quá trình huấn luyện trên toàn bộ tập dữ liệu huấn luyện.
  - ***learning_rate = 2e-5***: hệ số học của mô hình.
  - ***per_device_train_batch_size = batch_size***: kích thước của mỗi lô dữ liệu trên mỗi GPU. Trong trường hợp này, mỗi GPU sẽ xử lý một lô dữ liệu có kích thước tối đa là 64 mẫu.

+ ***per_device_eval_batch_size = batch_size***: kích thước của mỗi lô dữ liệu trong quá trình đánh giá mô hình trên mỗi GPU.
+ ***weight_decay = 0.01***: hệ số giảm trọng lượng L2 được sử dụng để điều chỉnh quá trình học của mô hình. Hệ số này giúp tránh hiện tượng overfitting trong quá trình huấn luyện.
+ ***evaluation_strategy = 'epoch'***: chiến lược đánh giá mô hình trong quá trình huấn luyện. Trong trường hợp này, mô hình sẽ được đánh giá sau mỗi epoch.
+ ***load_best_model_at_end = True***: cờ chỉ định xem mô hình có được tải lại từ checkpoint tốt nhất (checkpoint có độ đo đánh giá tốt nhất) sau khi huấn luyện hoàn tất.
+ ***save_strategy = "epoch"***: đây là chiến lược lưu trữ các checkpoint của mô hình trong quá trình huấn luyện. Trong trường hợp này, các checkpoint sẽ được lưu trữ sau mỗi epoch.

– Huấn luyện: Dựa theo mô hình pre-trained, tiếp tục train cho dữ liệu nhóm sử dụng với, các tham số đã thiết lập trước đó:
+ ***model*** và ***tokenizer*** dựa trên mô hình pre-trained sử dụng trước đó
+ ***args=training_args***: đưa các tham số đã thiết lập ở trên vào mô hình.
+ ***compute_metrics=compute_metrics***: sử dụng độ đo F1-score cho mô hình.
+ ***train_dataset=tokenized_sentence['train']***: sử dụng tokenized_sentence['train'] đã được mã hoá về Tensor để huấn luyện.
+ ***eval_dataset=tokenized_sentence['validation']***: sử dụng tokenized_sentence['validation'] đã được mã hoá về Tensor để đánh giá hiệu suất của mô hình trong mỗi lần chạy.

3.3 Đánh giá mô hình

– Nhóm đánh giá mô hình dựa trên độ đo **F1-score:**

$$F1\ score = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

+ Trong đó:

$$Precision = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

```
target_names = ['negative','neutral','positive']
print(classification_report(df_test['sentiment'], y_test_pred_labels))
```

```
              precision    recall  f1-score   support

           0       0.91      0.88      0.90      1154
           1       0.89      0.95      0.92      1685
           2       0.95      0.90      0.92      1284

    accuracy                           0.91      4123
   macro avg       0.92      0.91      0.91      4123
weighted avg       0.92      0.91      0.91      4123
```

Classification_report của mô hình sau khi trên train của nhóm

- Nhận xét:
  + Mô hình có độ chính xác (accuracy) trên tập dữ liệu test là 0.91, có nghĩa là mô hình dự đoán đúng khoảng 91% các trường hợp trong tập dữ liệu test.
  + Trong số ba lớp được dự đoán, lớp 1 (Neutral) có độ recall cao nhất nhưng ngược lại, lại có precision nhỏ nhất, có nghĩa là mô hình đang phân loại nhiều điểm vào lớp đó và có thể gây ra những sai sót trong việc quyết định. Lớp 0 (Negative) có độ recall thấp nhất (0.88), có nghĩa là mô hình dễ nhầm lẫn các trường hợp thuộc lớp này với lớp khác.
  + Từ các giá trị precision, recall và f1-score của mô hình đều khá cao và ở mức tương đương, cho thấy khả năng dự đoán của mô hình đồng đều trên các lớp.

⇨ Là mô hình tốt.

4. Web app

4.3 Thư viện sử dụng: streamlit

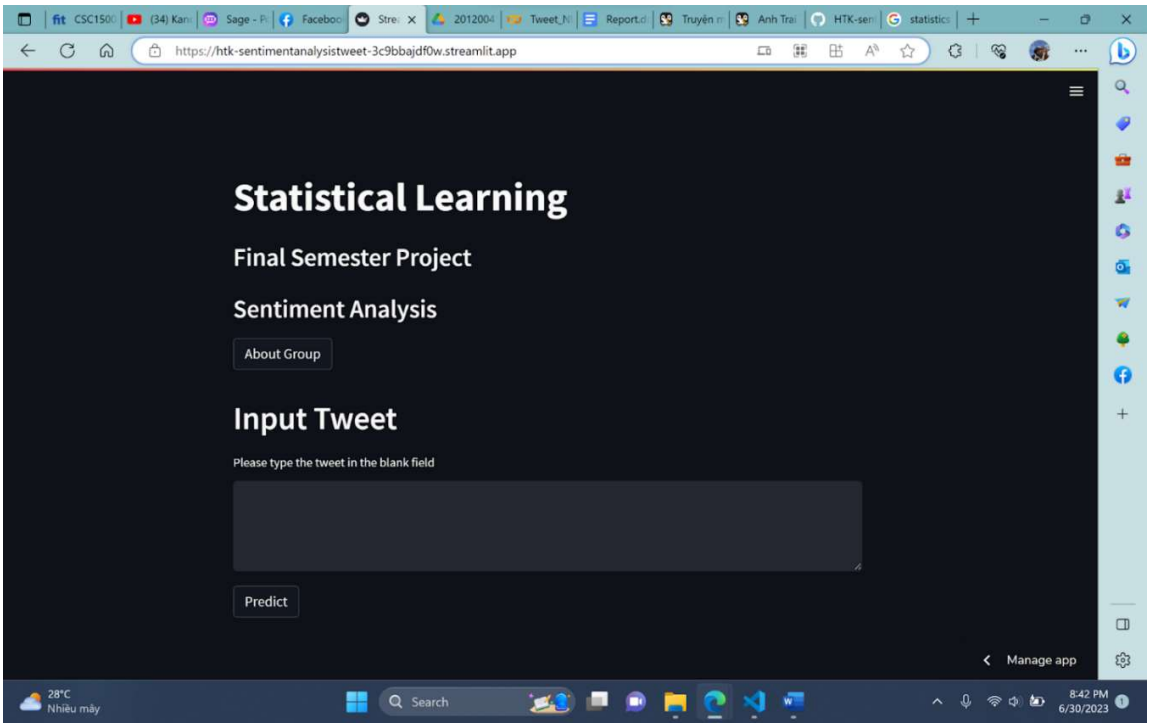
- Streamlit là một thư viện Python mã nguồn mở, giúp dễ dàng xây dựng các ứng dụng web tương tác cho các dự án học máy và khoa học dữ liệu. Streamlit cho phép người dùng thiết kế các ứng dụng web tương tác cho các mô hình học máy, trình bày các tập dữ liệu, và thực hiện các tác vụ phân tích dữ liệu một cách trực quan và dễ dàng.

- Streamlit có thể được sử dụng để xây dựng các ứng dụng web cho các mô hình học máy, các báo cáo phân tích dữ liệu, các trình trình bày kết quả động, các ứng dụng dự đoán dữ liệu, và nhiều ứng dụng khác. Streamlit có thể hoạt động với các framework học máy phổ biến như TensorFlow, PyTorch, và Scikit-learn.

Logo thư viện Streamlit

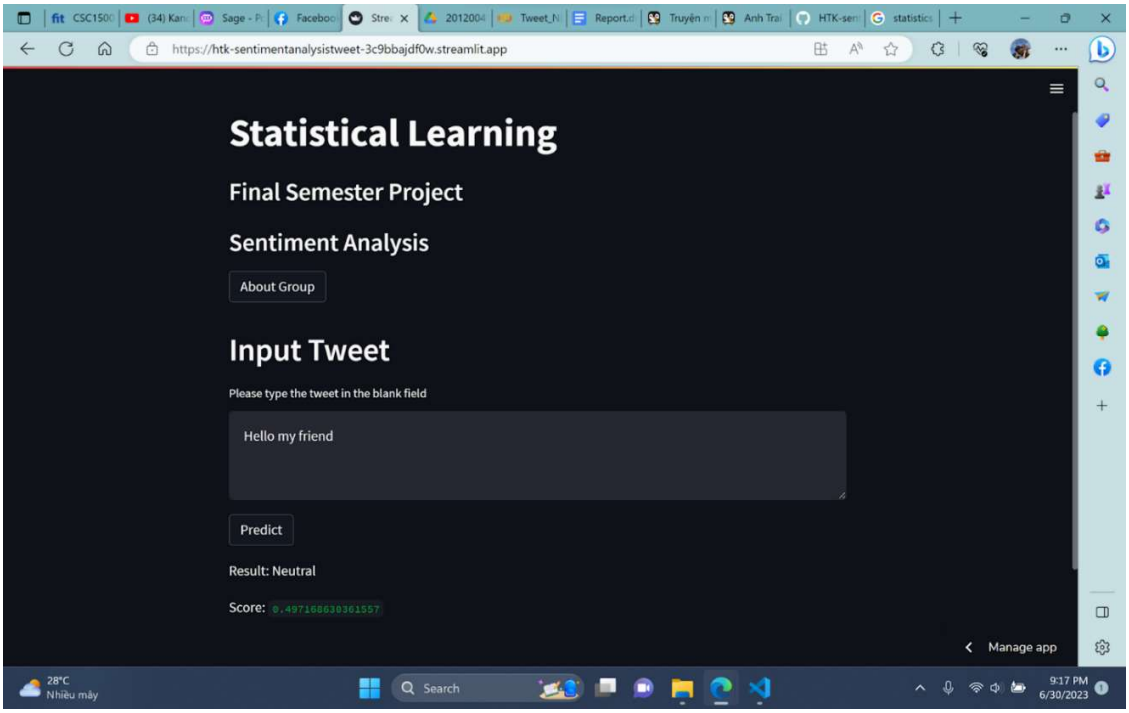
4.4 Giới thiệu web[4]:

- Link: htk-sentimentanalysistweet-3c9bbajdf0w.streamlit.app
  - + Nút "About Group": Xem thông tin thành viên nhóm.
  - + Ô nhập dữ liệu: Người dùng nhập câu tweet cần dự đoán.
  - + Nút "Predict": Dùng để dự đoán tweet đã nhập.



Giao diện web

- Demo: Nhập câu tweet "Hello my friend" vào trường nhập dữ liệu và nhấn "Predict". Kết quả thu được sẽ được in ra bên dưới, trong hình kết quả là "Neutral" (trung lập).



Hình ảnh demo của nhóm

C. Tham khảo:

[1]:https://www.kaggle.com/datasets/yasserh/twitter-tweets-sentiment-dataset

[2]:https://creativecommons.org/publicdomain/zero/1.0/

[3]: https://viblo.asia/p/bert-buoc-dot-pha-moi-trong-cong-nghe-xu-ly-ngon-ngu-tu-nhien-cua-google-RnB5pGV7lPG

https://phamdinhkhanh.github.io/2020/05/23/BERTModel.html#14-ti%E1%BA%BFp-c%E1%BA%ADn-n%C3%B4ng-v%C3%A0-h%E1%BB%8Dc-s%C3%A2u-trong-%E1%BB%A9ng-d%E1%BB%A5ng-pre-training-nlp

[4]: https://www.youtube.com/watch?v=D1V2oZTc3K8