

BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC UEH
TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ



ĐỒ ÁN MÔN HỌC

ĐỀ TÀI:

**XÂY DỰNG MÔ HÌNH PHÂN TÍCH CẢM XÚC
NGƯỜI DÙNG VỀ ĐẠI HỌC UEH DỰA TRÊN
ĐÁNH GIÁ TỪ GOOGLE MAPS**

Học phần: XỬ LÝ NGÔN NGỮ TỰ NHIÊN

Nhóm sinh viên:

1. LÊ THỊ NGỌC ÁNH
2. TRẦN PHẠM HẢI NAM
3. LÝ MINH NGUYỄN

Chuyên ngành: KHOA HỌC DỮ LIỆU

Khóa: K47

Hệ: CHÍNH QUY

Giảng viên: T.S. ĐẶNG NGỌC HOÀNG THÀNH

TP. Hồ Chí Minh, ngày 21 tháng 12 năm 2023

LỜI CẢM ƠN

Kính gửi **Thầy Đặng Ngọc Hoàng Thành**,

Lời đầu tiên, nhóm xin gửi lời cảm ơn chân thành đến thầy về sự hướng dẫn tận tâm trong suốt học phần lập trình xử lý ngôn ngữ tự nhiên. Những kiến thức mà Thầy truyền đạt không chỉ giúp nhóm hiểu sâu về lý thuyết mà còn là nền tảng quan trọng để nhóm có thể tự tin nghiên cứu và tiếp cận các kiến thức mới.

Đồ án cuối kỳ của nhóm không chỉ là cơ hội để áp dụng những kiến thức đã học mà còn là thách thức để phát triển khả năng tư duy và giải quyết vấn đề. Nhóm rất biết ơn vì Thầy đã tạo ra một môi trường học tập tích cực và chủ động.

Cuối cùng, nhóm xin bày tỏ lòng biết ơn về sự tận tâm về kiến thức truyền dạy của Thầy cũng như sự cảm thông của Thầy với hoàn cảnh của nhóm khi đã cho nhóm thêm thời gian để hoàn thành tốt bài đồ án. Những kiến thức quý báu mà Thầy truyền đạt sẽ là bước đệm quan trọng cho quá trình học tập, nghiên cứu của nhóm và là nguồn kinh nghiệm quý báu để tiếp tục phát triển tốt hơn các bài đồ án sau này.

Trân trọng cảm ơn thầy,

Nhóm nghiên cứu.

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI.....	1
1. Giới thiệu về đề tài.....	1
2. Mục tiêu nghiên cứu	1
3. Phương pháp nghiên cứu.....	1
4. Phương pháp thực hiện	1
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	2
1. Khái quát về Phân tích cảm xúc.....	2
2. Tổng quan lý thuyết các mô hình.....	2
a. Mô hình học máy: <i>Naïve Bayes</i>	2
b. Mô hình học máy: <i>Maxent</i>	3
c. Mô hình học sâu: <i>Convolutional Neural Network</i>	4
d. Mô hình học sâu: <i>Bidirectional Long-Short Term Memory</i>	5
CHƯƠNG 3: TỔNG QUAN BỘ DỮ LIỆU.....	7
1. Sơ lược bộ dữ liệu.....	7
2. Phương pháp trích xuất dữ liệu	7
3. Mô tả thuộc tính bộ dữ liệu.....	10
CHƯƠNG 4: KHÁM PHÁ VÀ PHÂN TÍCH DỮ LIỆU.....	11
1. Phân tích khám phá dữ liệu (Exploratory Data Analysis - EDA)	11
2. Tiền xử lý dữ liệu.....	12
5. POS tagging	15
a. Thực hiện gán nhãn phân loại	15
CHƯƠNG 5: HUẤN LUYỆN DỮ LIỆU DỰA TRÊN CÁC MÔ HÌNH	23
1. Huấn luyện mô hình.....	23
a. Sử dụng <i>Naïve Bayes</i>	23
b. Sử dụng <i>MaxEnt</i>	24
c. Sử dụng <i>CNN</i> và <i>BiLSTM</i>	27
2. Đánh giá so sánh trực quan	33
3. Ứng dụng kết quả (Demo)	34
CHƯƠNG 6: KẾT LUẬN VÀ ĐÁNH GIÁ.....	39
1. Kết quả đạt được	39
2. Hạn chế	39
3. Các phương pháp cải tiến đề xuất	39

LỜI MỞ ĐẦU

Ngôn ngữ tự nhiên là một trong những phương tiện giao tiếp mạnh mẽ nhất của con người và nó ngày càng trở thành trung tâm của sự quan tâm trong lĩnh vực công nghệ. "Xử lý Ngôn ngữ tự nhiên" không chỉ đánh dấu một bước tiến quan trọng trong sự hiểu biết về cách con người tương tác với máy tính mà còn là cơ hội để chúng ta tận dụng sức mạnh của ngôn ngữ tự nhiên trong việc giải quyết các vấn đề và nhiệm vụ phức tạp.

Nhóm sẽ bắt đầu hành trình này bằng việc khám phá khả năng của ngôn ngữ tự nhiên, từ cơ bản đến những ứng dụng tiên tiến trong lập trình. Mục tiêu của nhóm không chỉ là học cách sử dụng công cụ và thư viện, mà còn là hiểu rõ về tầm quan trọng của việc kết hợp khả năng sáng tạo và tính hiệu quả trong việc lập trình.

DANH MỤC VIẾT TẮT

UEH: Đại học UEH

EDA: Exploring Data Analysis

POS: Part-of-Speech

CNN: Convolutional Neural Network

BiLSTM: Bidirectional Long-Short Term Memory

CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

1. Giới thiệu về đề tài

Phân tích cảm xúc (Sentiment Analysis) là một lĩnh vực quan trọng trong xử lý ngôn ngữ tự nhiên, nơi tập trung vào việc hiểu và phân loại cảm xúc trong văn bản. Phương pháp này được ứng dụng rộng rãi trong việc đánh giá ý kiến người dùng, đo lường tình cảm trong bình luận và quản lý dư luận trực tuyến.

Trong đề tài này, nhóm tập trung vào phân tích cảm xúc trong những đánh giá về trường Đại học Kinh tế Thành phố Hồ Chí Minh (UEH). Dựa vào các thuộc tính của bộ dữ liệu, đề tài được xây dựng nhằm phân biệt những thông tin quan trọng về ý kiến và cảm nhận của người dùng đối với các khía cạnh như cơ sở vật chất, chất lượng giảng dạy hoặc yếu tố con người.

2. Mục tiêu nghiên cứu

Mục tiêu của đề tài không chỉ giúp hiểu rõ hơn ý kiến người dùng về UEH mà còn so sánh khả năng giải quyết của các thuật toán phân loại cảm xúc. Hướng phát triển có thể tập trung vào việc kết hợp thông tin từ nhiều thuật toán để tối ưu hóa khả năng dự đoán và cải thiện độ chính xác.

3. Phương pháp nghiên cứu

Phương pháp nghiên cứu được nhóm sử dụng đa dạng theo các bước như sau:

- Thu thập dữ liệu: Phương pháp Cào dữ liệu (Data crawling);
- Khám phá và chuẩn bị dữ liệu: Trực quan các loại biểu đồ, phương pháp xử lý dữ liệu tiếng Việt và tiếng Anh;
- Phân tích dữ liệu: Phân tích và đánh giá dữ liệu dựa trên POS tagging;
- Huấn luyện mô hình: Sử dụng các mô hình thuật toán Naive Bayes, Maxent, Convolutional Neural Network VÀ Bidirectional Long-Short Term Memory để đưa ra tỷ lệ dự đoán chính xác;
- Đánh giá kết quả: Trực quan và sử dụng thời gian và các chỉ số đánh giá khác như Precision, Recall, F1-score để đưa ra kết luận tốt nhất;
- Ứng dụng kết quả: Sử dụng html/CSS để triển khai micro web xây dựng demo.

4. Phương pháp thực hiện

Nhóm sử dụng các thuật toán để xác định độ chính xác cao nhất. Các bước của thuật toán được xây dựng bằng thư viện mã nguồn mở gồm các bước khởi tạo tham số, tiền xử lý, xây dựng mô hình, đánh giá mô hình và thử nghiệm một vài ví dụ để dự đoán.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

1. Khái quát về Phân tích cảm xúc

Bài toán Phân tích cảm xúc (hay Sentiment Analysis) thường tập trung vào tính phân cực của văn bản (tích cực hoặc tiêu cực), tuy nhiên, nâng cao hơn, bài toán này còn có thể sử dụng để phát hiện những cảm giác và cảm xúc cụ thể (tức giận, vui, buồn...), tính khẩn cấp (khẩn cấp, không khẩn cấp) và thậm chí cả ý định (sự quan tâm, không quan tâm). Tùy thuộc vào phương pháp diễn giải phản hồi và truy vấn của khách hàng, ta có thể xác định và điều chỉnh các danh mục để đáp ứng nhu cầu phân tích cảm tính của mình. Hiện nay có rất nhiều bài toán xoay quanh chủ đề này:

- Phân tích cảm xúc theo thang đo (Graded Sentiment Analysis): *sử dụng để diễn giải xếp hạng 5 sao trong một bài đánh giá, ví dụ như rất tích cực sẽ là 5 sao, còn rất tiêu cực sẽ tương ứng với 1 sao.*
- Phát hiện cảm xúc (Emotion Detection): *bài toán này vượt ra ngoài những bài toán phân cực truyền thống để phát hiện những khía cạnh cảm tính khác như hạnh phúc, thất vọng, tức giận và buồn bã...*
- Phân tích cảm xúc đa ngôn ngữ: *đây là bài toán phân tích khá phức tạp vì liên quan nhiều đến quá trình tiền xử lý và tài nguyên, nhưng hầu hết các tài nguyên đều có sẵn trực tuyến trong khi những tài nguyên khác cần được khởi tạo thủ công, ví dụ như văn bản đã dịch hoặc thuật toán phát hiện tiếng ồn.*

2. Tổng quan lý thuyết các mô hình

a. Mô hình học máy: Naïve Bayes

Naïve Bayes là một thuật toán phân lớp được mô hình hoá dựa trên định lý Bayes trong xác suất thống kê. Với sự lựa chọn sử dụng mô hình Naive Bayes là mô hình được sử dụng trong phân loại văn bản, đặc trưng đầu vào ở đây chính là tần suất xuất hiện của từ trong văn bản đó.

Định lý Bayes được sử dụng để xác định xác suất của một giả thuyết khi có sẵn kiến thức trước đó, nó phụ thuộc vào xác suất có điều kiện dựa trên công thức như sau.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Trong đó:

- $P(A|B)$ là xác suất hậu nghiệm, tức là xác suất của giả thuyết A khi sự kiện B xảy ra;

- $P(B|A)$ là xác suất khả năng, tức là xác suất của bằng chứng cho rằng giả thuyết A là đúng;
- $P(A)$ là xác suất trước, tức là xác suất của giả thuyết trước khi quan sát bằng chứng;
- $P(B)$ là xác suất cận biên, tức là xác suất của bằng chứng.

b. Mô hình học máy: Maxent

Bộ phân loại Maximum Entropy (MaxEnt) là một mô hình phân loại xác suất thuộc họ mô hình mũ, nó dựa trên nguyên lý của entropy tối đa, một khái niệm từ lý thuyết thông tin. Bộ phân loại MaxEnt thường được sử dụng trong xử lý ngôn ngữ tự nhiên và các nhiệm vụ máy học, đặc biệt là cho việc phân loại văn bản.

Bộ phân loại MaxEnt dựa trên nguyên lý Entropy tối đa, chọn mô hình có entropy lớn nhất từ tất cả các mô hình phù hợp với dữ liệu huấn luyện. Bộ phân loại MaxEnt mô hình phân phối xác suất qua các lớp dựa trên một tập hợp các đặc trưng quan sát được. Mô hình cơ bản của MaxEnt cho phân loại nhị phân là:

$$P(y|x) = \frac{1}{Z(x)} \exp \left[\sum_{i=1}^n \lambda_i f_i(x, y) \right]$$

Trong đó:

- $P(y|x)$ là xác suất của lớp y cho đầu vào x;
- $Z(x)$ là hàm phân loại đảm bảo rằng tổng xác suất các lớp bằng 1;
- λ_i là các đa thức Lagrange (tham số) liên quan đến các đặc trưng;
- f_i là các hàm đặc trưng ánh xạ đầu vào x và lớp y thành giá trị nhị phân chỉ ra sự có hoặc không có của một đặc trưng;

Hàm đặc trưng đóng vai trò quan trọng trong mô hình MaxEnt vì chúng giúp xác định mối quan hệ giữa các đặc trưng đầu vào và lớp đầu ra. Các đặc trưng này có thể chứa nhiều khía cạnh của dữ liệu đầu vào liên quan đến nhiệm vụ phân loại. Việc lựa chọn đặc trưng phụ thuộc vào vấn đề cụ thể được giải quyết.

Quá trình huấn luyện mô hình MaxEnt bao gồm ước lượng giá trị của các đa thức Lagrange (λ_i), thông thường, điều này được thực hiện bằng các kỹ thuật tối ưu hóa như Gradient Descent. Mục tiêu là tìm giá trị của tham số tối ưu hóa hàm log-likelihood của dữ liệu đào tạo quan sát được.

Khi mô hình MaxEnt được huấn luyện, nó có thể được sử dụng để đưa ra dự đoán trên dữ liệu mới, chưa được quan sát trước đó. Xác suất của mỗi lớp cho

trước các đặc trưng đầu vào được tính toán bằng cách sử dụng mô hình đã được huấn luyện.

c. Mô hình học sâu: *Convolutional Neural Network*

Mạng tích chập (Convolutional Neural Network - CNN) là một loại mạng nơ-ron nhân tạo được thiết kế đặc biệt để xử lý dữ liệu hình ảnh. CNN hoạt động bằng cách sử dụng các lớp tích chập để trích xuất các đặc trưng từ hình ảnh để phân loại hoặc thực hiện các nhiệm vụ xử lý hình ảnh khác.

Còn đối với việc xử lý ngôn ngữ tự nhiên, trong mô hình CNN, ma trận tích chập (convolutional matrix) dùng để lọc đầu vào văn bản, những đoạn văn bản này được tạo ra dựa trên số từ trong ma trận đầu vào, công việc này giúp tạo ra một hàm kích hoạt (activation function).

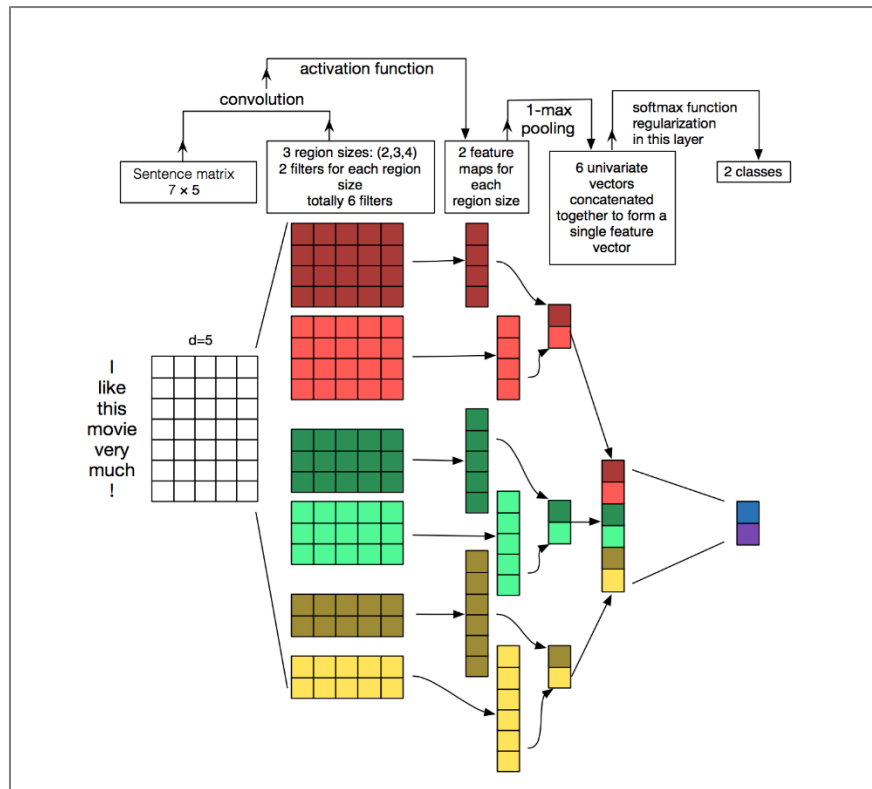
Theo bài nghiên cứu “*Sentiment analysis towards Jokowi's government using twitter data with convolutional neural network method*”, các tác giả cho rằng khi ta thực hiện quá trình tích chập (convolution) trên một ma trận có kích thước $N \times N$ bằng ma trận lọc (hoặc kernel) kích thước $m \times m$ với trọng số ω , kết quả mô hình sẽ trả về một ma trận mới, gọi là ma trận đầu ra tích chập (convolutional output matrix). Ma trận này có kích thước là $(N-m+1) \times (N-m+1)$. Bên cạnh đó, ta cũng thường áp dụng các hàm kích hoạt phi tuyến (non-linear activation) cho kết quả mới vừa được thu nhận này dựa trên công thức như sau.

$$x_{ij}^l = \sum_{a=0}^{m-1} \sum_{b=0}^{m-1} \omega_{ab} y_{(i+a)(j+b)}^{l-1}$$

Trong đó:

- x là kết quả ma trận của quá trình tích chập;
- m là chiều của ma trận tích chập;
- a, b là các chỉ số bắt đầu (index start) của ma trận;
- ω là trọng số được triển khai trong các layer ẩn;
- y là hàm tuyến tính để điều chỉnh giá trị số của ma trận.

Ý tưởng là trong quá trình tích chập, ma trận tích chập sẽ chia ma trận đầu vào thành các phần có kích thước khác nhau để tạo ra một bản đồ đặc trưng (feature mapping) cho ma trận đầu vào. Từ bản đồ đó, mô hình sẽ được kết nối thành một vector đặc trưng (feature vectors) trên các tầng nơ-ron. Cuối cùng từ các vector tìm được, ta sử dụng hàm softmax được để phân loại văn bản dựa trên ma trận đầu ra cuối cùng.



Hình 1: Mô phỏng quy trình xử lý ngôn ngữ tự nhiên dựa trên mô hình CNN (theo Dennybritz)

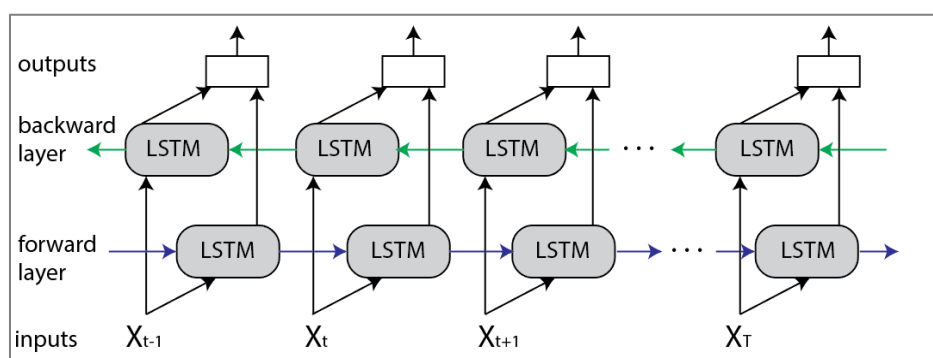
Để dễ hình dung hơn, nhóm sử dụng hình mô phỏng trong bài báo cáo “*A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification*”. Quy trình đang miêu tả ba kích thước vùng lọc (filter region size) là 2 (màu vàng), 3 (màu xanh) và 4 (màu đỏ), trong đó mỗi vùng có 2 bộ lọc (filter). Mỗi bộ lọc thực hiện tích chập trên ma trận câu (sentence matrix) để tạo ra các bản đồ đặc trưng có độ dài biến đổi. Sau đó, ta thực hiện cho từng bản đồ này đi qua lớp 1-max pooling để chọn số lớn nhất từ mỗi bản đồ.

Như vậy, ta sẽ có một vector đặc trưng đơn biến (univariate feature vector) từ cả 6 bản đồ. Ở tầng kế cuối, ta nối 6 đặc trưng này để tạo thành một vector đặc trưng mới, và tầng cuối cùng sử dụng hàm kích hoạt softmax để nhận vector mới này làm đầu vào và sử dụng nó để phân loại cho câu. Tác giả giả định rằng bài toán ở đây là phân loại nhị phân (binary classification) giống với bài toán mà nhóm sắp thực hiện, do đó đầu ra cuối cùng chỉ có 2 phần tử. Lưu ý rằng, activation nhóm tác giả sử dụng là “softmax”, ngoài ra, ta có thể sử dụng nhiều hàm kích hoạt khác ví dụ như “sigmoid” cho Logistic hoặc “ReLU” (Rectified Linear Activation).

d. Mô hình học sâu: Bidirectional Long-Short Term Memory

Tương tự với CNN, nhóm cũng kết hợp sử dụng mô hình Bidirectional Long Short-Term Memory (BiLSTM). Mô hình này là một mạng nơ-ron hồi quy sử dụng chủ yếu trong lĩnh vực xử lý ngôn ngữ tự nhiên, khác với mô hình LSTM thông thường, mô hình này xử lý đầu vào ở cả hai hướng từ đầu đến cuối và ngược lại. Điều này giúp nó có khả năng sử dụng thông tin từ cả hai phía của dữ liệu đầu vào.

Để làm được điều này, BiLSTM sẽ thêm một layer LSTM nữa để chạy ngược lại, sau đó, mô hình sẽ kết hợp các đầu ra của cả hai layer LSTM bằng cách sử dụng các đại lượng như trung bình hoặc tổng.



Hình 2: Lưu đồ quy trình xử lý dựa trên mô hình BiLSTM (theo Baeldung)

BiLSTM nổi bật với khả năng xử lý thông tin từ cả hai hướng của chuỗi, giúp mô hình hiểu thứ tự ràng buộc của câu. Sự linh hoạt trong cách kết hợp đầu ra này cũng có thể tạo ra khả năng tùy chỉnh cao trong việc học và đối phó với các loại ngôn ngữ phức tạp.

CHƯƠNG 3: TỔNG QUAN BỘ DỮ LIỆU

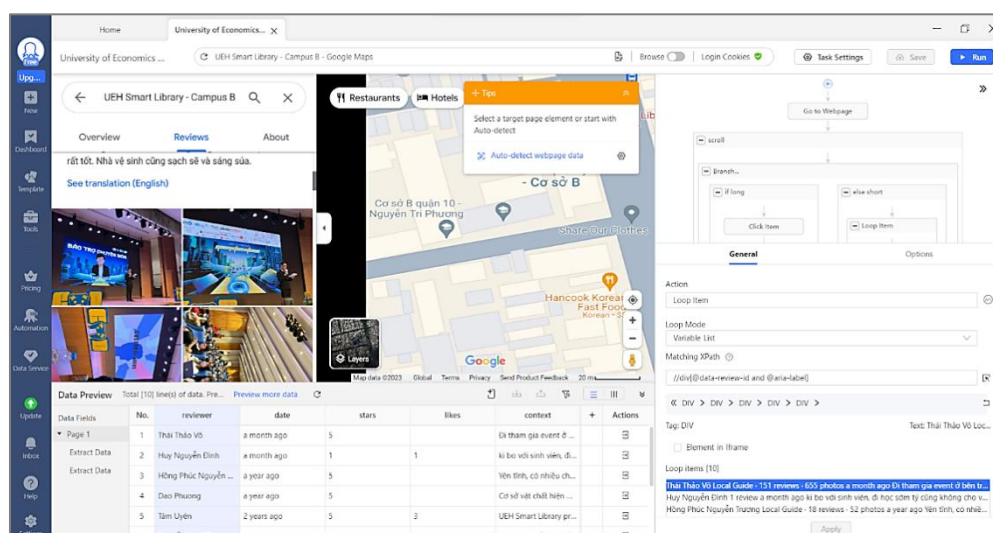
1. Sơ lược bộ dữ liệu

Bộ dữ liệu nhóm sử dụng có nội dung liên quan tới chính trường đại học mà các thành viên của nhóm đang theo học. Bộ dữ liệu này được nhóm tổng hợp từ các tập dữ liệu nhỏ hơn của các cơ sở trực thuộc quản lý của nhà trường, những tập nhỏ hơn này được nhóm thực hiện cào dữ liệu (data crawling) trực tiếp từ trên nền tảng Google Maps vào ngày 10 tháng 11 năm 2023. Vào thời điểm nhóm thực hiện trích xuất dữ liệu và tổng hợp, bộ dữ liệu tổng hợp này bao gồm khoảng 2400 quan sát, trong đó có khoảng 900 quan sát có bình luận đánh giá.

Với mục tiêu phân tích và khám phá những điểm cần cải thiện của trường dựa trên những bình luận tiêu cực, nhóm đã sử dụng tổng cộng 7 biến thuộc tính, bao gồm *reviewer*, *date*, *stars*, *likes*, *context*, *location*, *sentiment*. Chi tiết ý nghĩa các thuộc tính và phương pháp trích xuất dữ liệu sẽ được nhóm cụ thể hóa ở các phần tiếp theo.

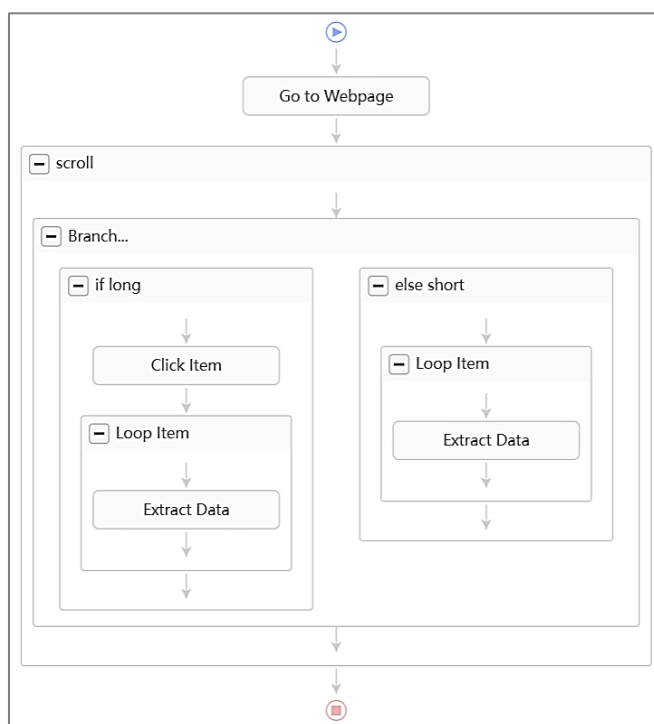
2. Phương pháp trích xuất dữ liệu

Ban đầu để thực hiện trích xuất dữ liệu, nhóm dự định sử dụng *selenium* hoặc những trang web hỗ trợ như *outscaper*, cuối cùng, nhóm quyết định tham khảo cách sử dụng phần mềm Octoparse của kênh Youtube *François from Octoparse* để thực hiện công việc trích xuất.



Hình 3: Giao diện tổng quan trích xuất dữ liệu bằng phần mềm Octoparse

Bởi vì nhóm chưa có kinh nghiệm trong việc cào dữ liệu nên đầu tiên, nhóm thực hiện tổng hợp tất cả các đường dẫn Google Maps liên quan tới các cơ sở của trường. Sau đó, nhóm thực hiện cho chạy tự động việc cào mỗi địa điểm với quy trình trích xuất xác định. Cuối cùng, nhóm sử dụng python để kết hợp tất cả các bộ dữ liệu thu được và gán nhãn địa điểm để phân biệt.

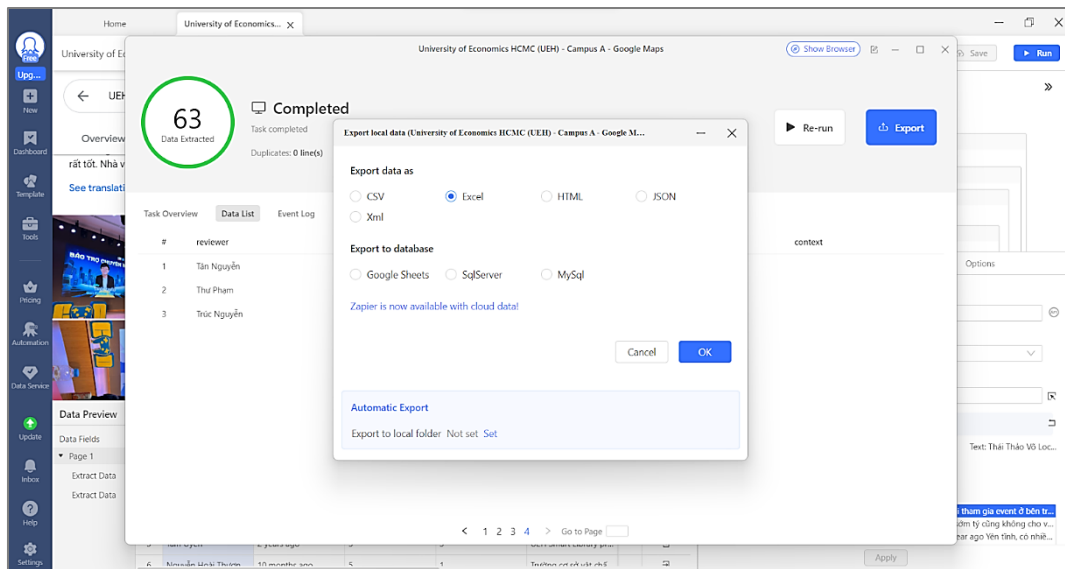


Hình 4: Cây quy trình trích xuất dữ liệu thực hiện bằng Octoparse

Để xây dựng được cây quy trình, nhóm cần phải xác định trước phương pháp thực hiện theo từng bước bắt đầu từ việc mở đường dẫn được cung cấp sẵn cho tới việc trích xuất dữ liệu cụ thể bằng XPath - một dạng đường dẫn sử dụng ngôn ngữ XML để tìm kiếm các element. Chi tiết quy trình được nhóm xác định như sau:

- Bước 1: Truy cập vào trang web bằng đường dẫn được nhóm lấy theo Google Maps từ trước;
- Bước 2: Sử dụng XPath để tìm thanh scroll bar trong tab “Reviews”, sau đó, chạy vòng lặp để thực hiện kéo thanh này đến khi tab này không còn hiện thêm thông tin mới nào;
- Bước 3: Xây dựng một nhánh của cây quy trình này thành câu lệnh có điều kiện if-else;
- Bước 4: Sử dụng XPath để tìm kiếm element “More”. Nếu có, bài đánh giá sẽ được xem là một bài viết dài và element đó sẽ được kích hoạt. Tất cả những dữ liệu liên quan sẽ được trích xuất xuống toàn bộ và lưu vào *Data Fields*;

- Bước 5: Nếu không có element “*More*” ở trong bài đánh giá, đây sẽ xem như một bài viết ngắn và bỏ qua bước kích hoạt dư thừa. Tất cả dữ liệu liên quan cũng sẽ được cào và lưu tiếp vào Data Fields;
- Bước 6: Sau khi vòng lặp chạy hết toàn bộ nội dung đánh giá, trong vòng 10 giây tiếp theo nếu như không có bất cứ sự thay đổi nào diễn ra, quá trình trích xuất sẽ được kết thúc và dữ liệu trong *Data Fields* sẽ được hiển thị ra màn hình.



Hình 5: Giao diện kết quả trích xuất dữ liệu với địa điểm “Thư viện cơ sở B của trường”

Quy trình này tương đối dễ hiểu, duy nhất hai điểm khó khăn của nhóm khi thực hiện công việc trích xuất này là bước lấy chính xác đường dẫn trang web đánh giá yêu cầu và bước lấy XPath, có thể bởi vì nhóm chưa có kinh nghiệm cào dữ liệu cũng như chưa từng có trải nghiệm với việc sử dụng ngôn ngữ XML trước đó.

Bộ dữ liệu sau khi được cào xuống đã được nhóm sử dụng python để thực hiện nối các tập dữ liệu riêng lẻ và gán vào thêm 2 cột thuộc tính khác là “*location*” và “*sentiment*”, với cột “*location*” được trích xuất từ chính cơ sở địa điểm mà nhóm thu thập bài đánh giá của cơ sở đó.

3. Mô tả thuộc tính bộ dữ liệu

Sau khi thu thập dữ liệu, nhóm thực hiện mô tả các thuộc tính bằng bảng sau đây.

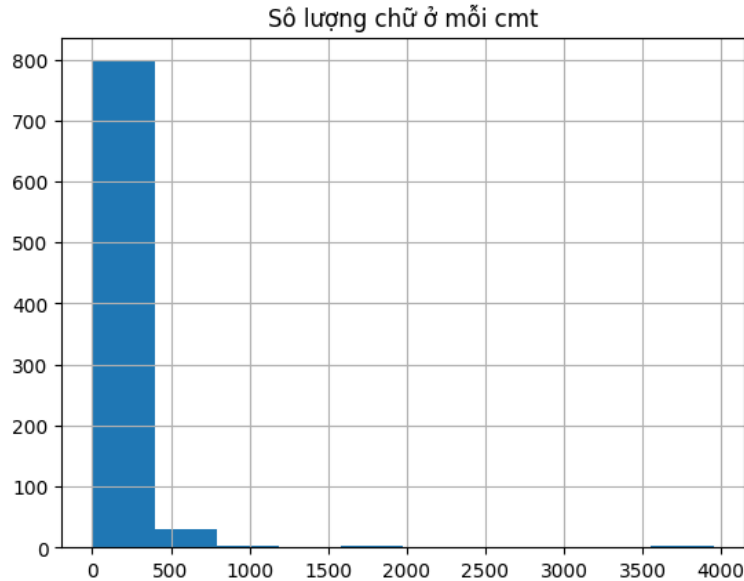
TÊN THUỘC TÍNH	MÔ TẢ	CHÚ THÍCH
reviewer	Tên của người đánh giá dựa theo tài khoản Google	Những tên này có thể không tuân theo quy luật bình thường
date	Thời điểm người đánh giá đăng bài đánh giá	Thời gian từ lúc bắt đầu đánh giá tới thời điểm ngày 10/11/2023
stars	Số lượng sao người đánh giá đăng trong bài đánh giá	Mức độ hài lòng đối với địa điểm đó
likes	Số lượt thích của đánh giá	Mức độ đồng cảm (độ tin cậy) đối với nội dung đánh giá
context	Nội dung bài đánh giá	Bao gồm tiếng Anh và tiếng Việt
location	Địa điểm đánh giá đề cập trong bài viết	Đây là một trong các cơ sở của UEH
sentiment	Cảm xúc của bài đánh giá	Có 2 giá trị là “positive” (tích cực) và “negative” (tiêu cực)

Bảng 1: Bảng mô tả các thuộc tính trong bộ dữ liệu

CHƯƠNG 4: KHÁM PHÁ VÀ PHÂN TÍCH DỮ LIỆU

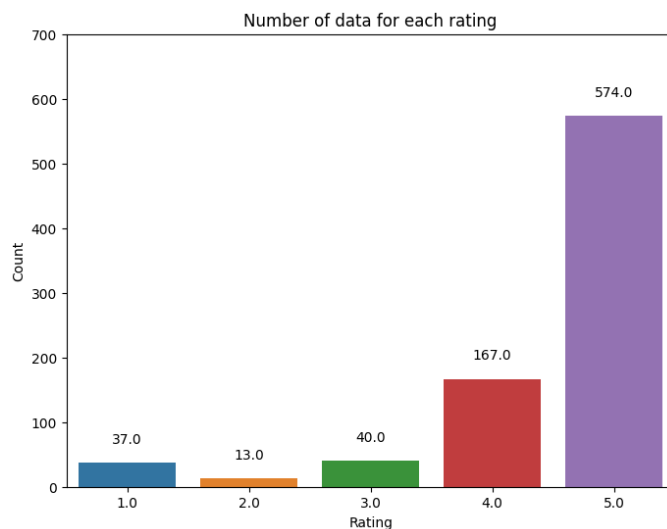
1. Phân tích khám phá dữ liệu (Exploratory Data Analysis - EDA)

Đầu tiên, nhóm tiến hành khám phá dữ liệu, loại bỏ các hàng bị trùng lặp. Sau đó, tìm hiểu số lượng chữ trong mỗi bình luận.



Hình 6: Biểu đồ cột thể hiện số lượng chữ ở mỗi câu bình luận

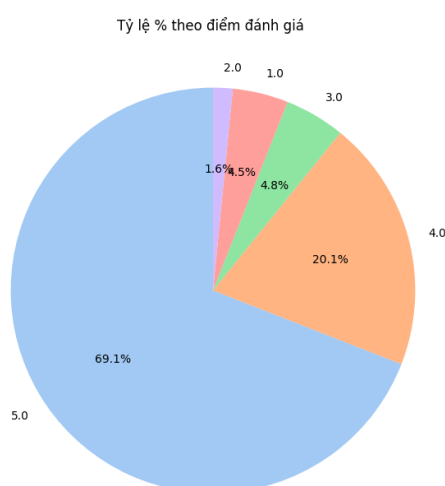
Khi nhận thấy số lượng chữ trong mỗi bình luận dưới 1000, tiến hành thống kê tìm hiểu số lượng và các dòng có số lượng chữ lớn hơn mức 500. Kết quả cho thấy tổng số dòng có số từ trên 500 là 5 dòng. Các dòng có số từ nhiều như vậy chủ yếu nói về lịch sử trường và các thành tích của trường, và tư hào về thành tích của trường. Tiếp theo, nhóm tiến hành tìm hiểu thông tin về cột khác là điểm đánh giá (stars).



Hình 7: Biểu đồ cột thể hiện số lượng câu bình luận theo điểm đánh giá

Dựa vào hình này, nhóm nhận thấy có 574 người đánh giá 5 sao và chiếm ưu thế so với các đánh giá còn lại, tiếp theo là 167 người đánh giá 4 sao. Điều này cho thấy mức độ hài lòng và cảm xúc tích cực của người dùng cao. Với đánh giá 1 sao nhiều

hơn 2 sao, nhóm cần xem xét lý do vì sao có người dùng để lại bình luận với cảm xúc tệ nhất nhiều hơn gần gấp 3 so với việc cảm thấy hơi tệ. Nhóm cũng có thể nhìn xem mức độ phân bố rõ hơn theo biểu đồ tròn.



Hình 8: Biểu đồ tròn thể hiện tỷ lệ các câu bình luận theo điểm đánh giá

Từ biểu đồ, nhóm có thể thấy lượt đánh giá 5 sao chiếm gần 70% số đánh giá, cho thấy phản ứng tổng quát các cơ sở của trường là rất tích cực.

2. Tiền xử lý dữ liệu

Nhóm tiến hành tiền xử lý theo các bước như sau:

- **Chuyển đổi chữ cái thành chữ thường:** Đầu tiên, nhóm chuyển đổi tất cả các chữ cái trong văn bản thành chữ thường. Bước này giúp đảm bảo sự nhất quán và không phụ thuộc vào việc chữ cái được viết hoa hay thường, từ đó giảm độ phức tạp của dữ liệu và tối ưu hóa quá trình phân tách từ.
- **Xử lý dấu câu, emoji và biểu tượng:** Tiếp theo, nhóm tiến hành xử lý dấu câu, emoji, và các biểu tượng đặc biệt. Các ký hiệu này thường không mang lại nhiều thông tin ý nghĩa trong ngữ cảnh phân loại và có thể tạo nhiễu. nhóm có thể thay thế chúng bằng khoảng trắng hoặc loại bỏ chúng khỏi văn bản.
- **Thay thế ký tự đặc biệt:** Các ký tự đặc biệt thường không mang nhiều thông tin ý nghĩa trong ngữ cảnh dữ liệu này. nhóm quyết định thay thế chúng bằng khoảng trắng để loại bỏ lỗi và tăng tính tập trung vào các từ ngữ quan trọng.

Nhóm còn tiến hành loại bỏ các khoảng trắng dư thừa giúp đồng nhất hóa không gian giữa các từ. Đồng thời, loại bỏ khoảng trắng dư thừa cũng giúp tăng hiệu suất trong quá trình xử lý văn bản, vì các thao tác tiếp theo như tách từ, loại bỏ stopwords, hay đào tạo mô hình máy học sẽ được thực hiện trên dữ liệu đã được chuẩn hóa một cách đồng nhất. Điều này có thể giúp giảm thời gian và tài nguyên tính toán cần thiết cho các công đoạn xử lý và đào tạo mô hình

index	reviewer ▼	date	stars	likes	context	location	sentiment	pre_context	language
1	Vincent magnus Kenneth	2 years ago	5.0	NaN	Trường sạch đẹp, giảng viên dễ thương, chương trình học tốt - from Ueher 43	uehA	positive	trường sạch đẹp giảng viên dễ thương chương trình học tốt từ ueher 43	vi
3	Trà Lê Thị Thu	3 years ago	5.0	NaN	Ngôi trường có đội ngũ giáo viên giỏi, nhiệt tình và vô cùng dễ thương ♥ ...	uehA	positive	ngôi trường có đội ngũ giáo viên giỏi nhiệt tình và vô cùng dễ thương ...	vi
0	Thanh Nam Nguyễn	6 years ago	4.0	NaN	One of the best universities in Vietnam I m having great time at this place	uehA	positive	one of the best universities in vietnam i m having great time at this place	en
4	Long Trần	4 years ago	5.0	NaN	Hay tự ý đổi lịch học của học sinh :) nhưng nhìn chung rất là tốt gud gud	uehI	positive	hay tự ý đổi lịch học của học sinh nhưng nhìn chung rất là tốt gud gud	vi
2	Bình Phương Tang	a year ago	5.0	NaN	Cơ sở vật chất hiện đại, thật tuyệt vời. 1 trường Đại Học lớn của miền Nam	uehB	positive	cơ sở vật chất hiện đại thật tuyệt vời 1 trường đại học lớn của miền nam	vi

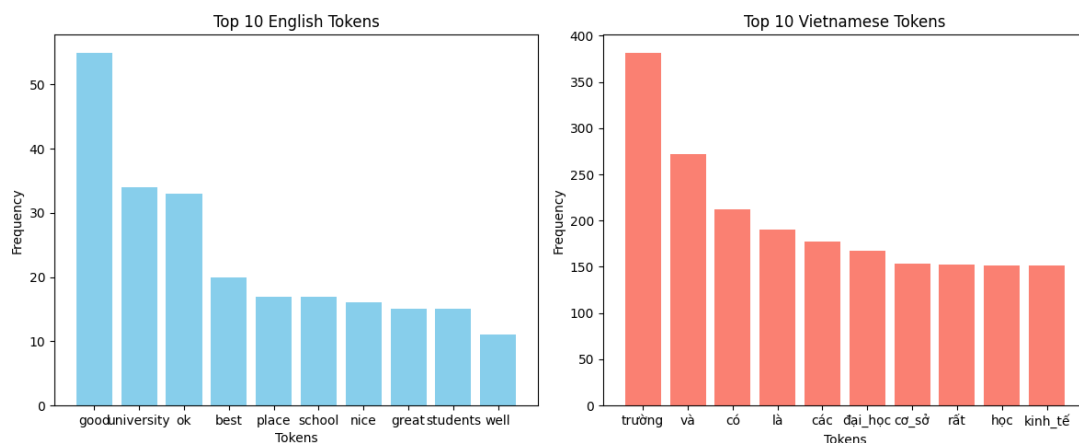
Hình 9: Bảng dữ liệu sau khi thực hiện tiền xử lý

Sau đó, nhóm thực hiện lưu lại file tiền xử lý cơ bản này vì khi chuyển đổi số thành chữ và stopwords, nhóm cần phân chia thành 2 dataframe theo 2 ngôn ngữ lần lượt là *vie_df* (theo tiếng Việt) và *eng_df* (theo tiếng Anh). Việc thực hiện chuyển số thành chữ, stopwords cho dataframe thành 2 dataframe mới sẽ giúp giảm các chữ nhiễu trong dữ liệu cũng như xử lý phần lớn chữ số thành chữ để thống kê.

3. Word Tokenize

Tách từ (word tokenization) là quá trình chia một đoạn văn bản thành các từ hoặc các “token” riêng lẻ. Mục tiêu là phân đoạn một câu hoặc một đoạn văn bản thành các thành phần từ để phục vụ cho quá trình chuyển đổi từ sang các vectơ tham chiếu (word2vec), từ đó áp dụng được các mô hình học máy thực hiện bài toán dự đoán.

Phương pháp tách từ thông dụng nhất chính là dựa vào khoảng trắng, tuy nhiên do tính mơ hồ và các trường hợp đặc biệt của ngôn ngữ, nhóm cần một phương pháp tiên tiến hơn là sử dụng mô hình ngôn ngữ được huấn luyện trước hoặc các phương pháp thống kê để tính toán tần suất xuất hiện của một số chuỗi ký tự hay từ. Ở đây, nhóm sẽ sử dụng hai mô hình tách từ có sẵn từ hai thư viện chuyên xử lý ngôn ngữ tự nhiên là *nlTK* và *pyvi*.



Hình 10: Biểu đồ cột thể hiện 10 token phổ biến nhất của bộ dữ liệu theo từng ngôn ngữ

Kết quả tách từ cho thấy tần suất xuất hiện của các từ tiếng Việt chiếm ưu thế hơn hẳn so với các từ vựng tiếng Anh, điều này cho thấy được sự vượt trội của các dòng đánh giá bằng tiếng Việt trong bộ dữ liệu, các từ xuất hiện thông dụng nhất trong

tiếng Anh là “good”, “university” và “ok”. Có thể thấy, hầu hết các từ xuất hiện nhiều nhất trong nhóm từ tiếng Anh mang xu hướng tích cực, trong khi đó, các từ có tần số xuất hiện nhiều nhất của trong nhóm từ tiếng Việt là “trường”, “và”, “có”, đây là những từ không miêu tả cụ thể cảm xúc rõ ràng.

4. Sentence Tokenize

Tách câu (sentence tokenization) là quá trình phân đoạn văn bản thành các câu riêng lẻ, mục tiêu quá trình này là nhận diện ranh giới giữa các câu trong đoạn văn bản. Phương pháp tách câu thông dụng nhất là dựa vào những ký tự dấu câu thường được đặt ở cuối câu hoặc dựa vào dấu cách nếu trong trường hợp không có dấu câu hỗ trợ.

Ngoài ra, có một số phương pháp tiên tiến sử dụng những mô hình ngôn ngữ được huấn luyện trước và phương pháp thống kê để xử lý các trường hợp đặc biệt của câu từ. Ở đây, nhóm sẽ sử dụng hai mô hình của hai thư viện *nltk* và *underthesea* để thực hiện tách câu cho cột context.

Nhóm sẽ thực hiện tách câu dựa trên bộ dữ liệu khác so với phân tách từ do các ký tự đặc biệt như các dấu chấm câu sẽ hỗ trợ tốt hơn cho các thư viện thực hiện tách câu. Tương tự như phân tách từ, nhóm xây dựng hàm tách dựa trên ngôn ngữ của cột context, ở đây với các dòng tiếng Việt nhóm sẽ áp dụng hàm *sent_tokenize* của thư viện *underthesea* và các dòng tiếng Anh là hàm *sent_tokenize* của thư viện *nltk*.

```
Top 5 Longest English Sentence Tokens:
I found that students did not adequately review material in the beginning of courses to build a solid foundation
Lenght: 344
-----
In first year in a science program, I would say I spent 2 nights per week socializing in first year, but otherwis
Lenght: 259
-----
Most of their office hours go unused for some reason, and then students complain about the difficulty of the mate
Lenght: 251
-----
Floor 7 has a photocopy machine (each student can print 500 pages for free in a semester if I remember correctly)
Lenght: 231
-----
The school often invites successful business owners and directors from big corporations to share their insights a
Lenght: 202
```

Hình 11: Kết quả 5 câu có độ dài lớn nhất trong tiếng Anh

```
Top 5 Longest Vietnamese Sentence Tokens:
Cùng với quá trình đổi mới và phát triển của đất nước, sau 45 năm hình thành và phát triển, với đội ngũ giáo sư
Lenght: 1360
-----
( K46 chia sẻ ) - Bảng điểm vẫn in Chữ Phân Hiệu khi đưa cho nhà tuyển dụng xem ( K46 chia sẻ ) - Giấy tạm hoãn c
Lenght: 654
-----
Đặc biệt, UEH vinh dự nằm trong Top 1000 Trường đào tạo kinh doanh tốt nhất thế giới (Theo BXH Eduniversal) từ t
Lenght: 622
-----
*Của cả KTX (dùng chung): • 1 phòng tự học, đầy đủ quạt, đèn và ổ cắm, kệ sách tham khảo; • Mỗi lầu đều có 1 máy
Lenght: 620
-----
Ngày 27.10.1976, là cột mốc quan trọng đánh dấu một bước chuyển mình mới cho giáo dục của Việt Nam - Một trường
Lenght: 572
```

Hình 12: Kết quả 5 câu có độ dài lớn nhất trong tiếng Việt

Kết quả cho thấy giống với phân tách từ khi các câu tiếng Việt có độ dài vượt trội so với các câu trong tiếng Anh. Điều này cho thấy các đánh giá trong tiếng Việt không những nhiều hơn về mặt số lượng mà nội dung trong từng dòng còn cung cấp cho người xem nhiều thông tin hơn về các cơ sở trực thuộc Đại học UEH.

5. POS tagging

a. Thực hiện gán nhãn phân loại

Để có cái nhìn chi tiết hơn về nội dung, nhóm quyết định tích hợp quá trình gán nhãn từ loại (Part-of-Speech tagging) vào việc phân tích. Mục tiêu của nhóm là khám phá cấu trúc ngôn ngữ của các nhận xét này, từ đó, phân loại các bình luận để xác định ý kiến của cộng đồng đối với từng lĩnh vực cụ thể cần cải thiện.

Bởi vì dữ liệu được cào dựa trên một nguồn mở, vậy nên các nội dung sẽ có thể được biểu diễn dưới nhiều loại ngôn ngữ khác nhau. Qua việc phân tích, nhóm nhận thấy số lượng bình luận tiếng Việt nhiều hơn gấp ba lần so với tiếng Anh. Chính vì vậy nhóm sẽ tách các bình luận này ra làm hai phần tương ứng với hai ngôn ngữ chính để thực hiện gán nhãn từ loại.

Đối với ngôn ngữ tiếng Việt, nhóm sử dụng thư viện `pyvi` để thực hiện xử lý văn bản và gán loại từ. Trong quá trình thực hiện, nhóm tạo dictionary để lưu trữ kết quả, mỗi từ trong nội dung văn bản sẽ được sắp xếp thành key-value, với key là từ được phân tách bằng hàm `ViTokenizer.tokenize()` còn value sẽ là nhãn từ loại tương ứng với token đó.

```
[('trường', 'NOUN'),  
 ('sạch', 'ADJ'),  
 ('đẹp', 'ADJ'),  
 ('giảng_viên', 'NOUN'),  
 ('dễ', 'ADJ')]
```

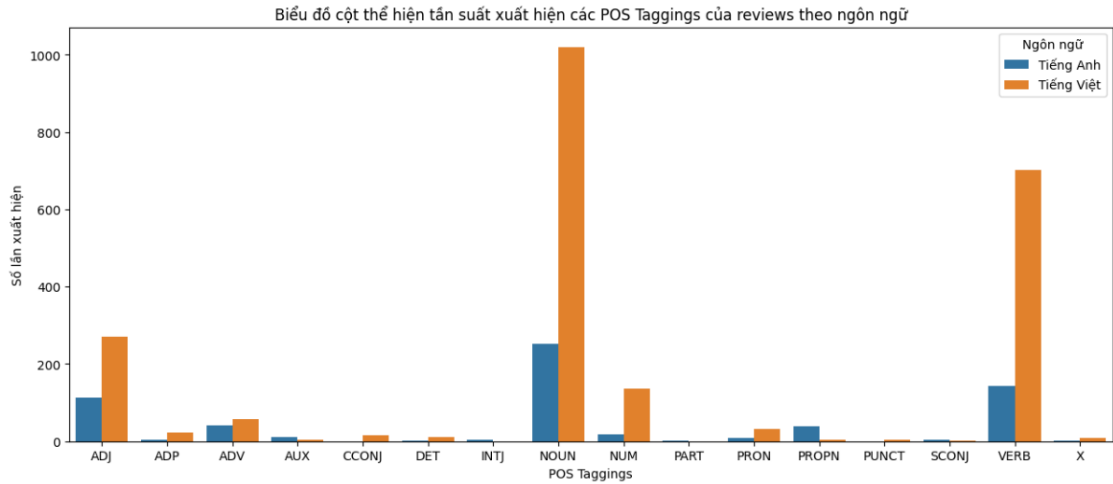
Hình 13: Đại diện một số kết quả POS Tagging cho các nhận xét tiếng Việt

Đối với ngôn ngữ tiếng Anh, bởi vì thư viện `pyvi` không thực hiện tốt đối với các ngôn ngữ khác nên nhóm sẽ sử dụng một thư viện khác là `spaCy` để xử lý cho các bình luận này. Ý tưởng thực hiện cũng tương tự với phương pháp tiếng Việt, ở đây, nhóm sẽ sử dụng mô hình `en_core_web_sm` để thực hiện, đây là một trong những mô hình pre-trained của thư viện để xử lý ngôn ngữ tự nhiên. Sau khi thực hiện, nhóm thu được kết quả và thực hiện kiểm tra bằng một số giá trị đại diện như sau.

```
[('one', 'NUM'),  
 ('best', 'ADJ'),  
 ('universities', 'NOUN'),  
 ('vietnam', 'PROPN'),  
 ('great', 'ADJ')]
```

Hình 14: Đại diện một số kết quả POS Tagging cho các nhận xét tiếng Anh

Sau khi thực hiện, nhóm thực hiện tổng hợp cả hai kết quả và tạo dataframe để trực quan.



Hình 15: Biểu đồ cột thể hiện tần suất xuất hiện loại từ của các bình luận

Qua biểu đồ trực quan, nhóm nhận thấy có một insights như sau:

- Bình luận tiếng Việt có tần suất xuất hiện nhiều hơn so với bình tiếng Anh, với số lượng tương ứng là 2658 (tiếng Việt) và 808 (tiếng Anh).
- Từ loại danh từ xuất hiện nhiều nhất trong cả hai ngôn ngữ, cho thấy rằng các đánh giá thường tập trung vào việc mô tả các tính năng hoặc đánh giá của các khu chức năng và dịch vụ của trường.
- Tính từ và trạng từ là các loại từ xuất hiện nhiều tiếp theo. Điều này cho thấy rằng các đánh giá có thể thường tập trung vào việc đưa ra đánh giá chất lượng, giá trị hoặc điểm cần cải thiện của các khu chức năng trong trường.

Những insights này rất hữu dụng để nhóm có thể hiểu rõ hơn về cách sử dụng ngôn ngữ người dùng trong các bình luận.

b. Phân tích bình luận tiêu cực

Để khai thác rõ hơn những giá trị công việc gán nhãn từ loại mang lại, nhóm tiến hành phân tích dựa trên những token loại danh từ xuất hiện nhiều nhất và các nhãn tiêu cực của bình luận chứa các token đó, từ đó, tìm những điểm cần cải thiện và thực hiện đề xuất.



Hình 16: Biểu đồ WordCloud các danh từ xuất hiện nhiều nhất

Đầu tiên, nhóm tiến hành trực quan bằng thư viện wordcloud và thu được kết quả như hình trên. Dựa vào biểu đồ nhóm nhận thấy rằng, một số token xuất hiện với tần suất nhiều nhất ở cả hai ngôn ngữ có thể liệt kê là “trường”, “cơ sở”, “sinh viên”, “không gian”, “giảng viên”... Vậy nên, nhóm sẽ thực hiện chia nội dung bình luận thành ba chủ đề chính:

- **Chủ đề “Giảng viên”:** bao gồm các bình luận về phương thức đào tạo của các giảng viên trên giảng đường hoặc những nhận xét về các phòng ban làm việc tại các cơ sở.
- **Chủ đề “Cơ sở vật chất”:** là các bình luận về chất lượng cơ sở vật chất trong các cơ sở, ví dụ như bãi xe, thư viện, phòng học, khuôn viên nhà trường...
- **Chủ đề “Hoạt động sinh viên”:** các bình luận về hoạt động câu lạc bộ/đội/nhóm của sinh viên tổ chức, ngoài ra, có thể bao gồm các nhận xét về ý thức và thái độ của người học xoay quanh những sự kiện, hoạt động diễn ra trong trường học.

Để có thể đề xuất những điểm cần cải thiện, nhóm tiến hành chỉ lọc ra những bình luận chưa tốt, hay còn gọi là những nhận xét gắn nhãn tiêu cực, để thực hiện phân tích dựa trên những bình luận này.

i. Chủ đề Giảng viên

Ở chủ đề này, nhóm thực hiện phân tích những bình luận tiêu cực xoay quanh về các giảng viên và phòng ban đào tạo của trường dựa trên bộ dữ liệu thu được. Trước khi tiến hành thực nghiệm, nhóm cần xác định rõ những từ khóa liên quan đến chủ đề này. Qua trực quan bằng biểu đồ mây từ loại và các lần kiểm tra tổng quan bộ dữ liệu, nhóm có thể liệt kê 9 từ khóa của chủ đề xuất hiện nhiều nhất

là “giảng viên”, “giáo viên”, “khoa”, “đội ngũ”, “cán bộ”, “quản lý”, “văn phòng”, “teacher”, “lecturer”.

Tiếp theo, nhóm thực hiện tìm những từ khóa này trong những nội dung bình luận tiêu cực và tiến hành sắp xếp theo thứ tự giảm dần của likes nhằm xác định độ tin cậy của bình luận đó dựa trên sự đồng cảm của những người bình luận khác. Kết quả trả về có tổng cộng 15 bình luận liên quan tới các từ khóa mà nhóm đã liệt kê.

	context	likes	location	sentiment	language
575	Thái độ văn phòng đào tạo k tốt. Mình vào hỏi ...	20.0	uehA	negative	vi
570	Tệ,mức học phí quá cao 930k/1 tín,tuyển sinh đ...	10.0	uehA	negative	vi
566	Một ngôi trường lừa đảo, lúc tuyển sinh thì nó...	9.0	uehI	negative	vi
568	Giảng viên không có tinh thần, phòng ban làm v...	9.0	uehA	negative	vi
550	Trường có cơ sở vật chất hiện đại không sử dụng...	7.0	uehB	negative	vi
557	NHƯỢC ĐIỂM:\n - Khi đi thực tập trường in cái ...	7.0	uehVL1	negative	vi
548	Cán bộ phòng ban hoàn toàn không có tin thần p...	6.0	uehA	negative	vi
530	Giảng viên không có tinh thần, phòng ban làm v...	4.0	uehB	negative	vi
499	Mình đã xác minh là chỗ này đã đập bỏ rồi mà g...	3.0	uehB	negative	vi
490	Cơ sở nhỏ xíu, cũ. Không có thư viện, không có...	2.0	uehE	negative	vi
352	ĐHKT TP.HỒ CHÍ MINH là cơ sở đào tạo đại học v...	1.0	uehB	negative	vi
591	Môi trường thân thiện, cơ sở chất lượng, giảng...	NaN	uehB	negative	vi
637	Cơ sở vật chất tốt, máy lạnh 🥶 gất lạnh\n Tuy ...	NaN	uehLibN	negative	vi
830	This is where I learned my economics degree. I...	NaN	uehA	negative	en
834	Prestigious and old school in town. Parking lo...	NaN	uehA	negative	en

Hình 17: Các bình luận tiêu cực về chủ đề Giảng viên

Bởi vì trang nguồn mà nhóm thực hiện cào và phân tích dữ liệu là một nguồn mở không giới hạn, vậy nên không thể tránh khỏi việc có những bình luận không liên quan đến trường hoặc những nhận xét kém tế nhị. Vậy nên, nhóm chỉ thực hiện chuẩn hóa và liệt kê 3 bình luận tiêu biểu trong chủ đề này để thực hiện việc phân tích.

- Tên người đánh giá:** NHU PHAM QUYNH;
Cơ sở UEH: Cơ sở B (279 Nguyễn Tri Phương, quận 10);
Nội dung: Môi trường thân thiện, cơ sở chất lượng, giảng viên tùy người: có tâm dễ hiểu hoặc giảng rất có tâm nhưng không hiểu, hoặc không hiểu gì. Trường quá nhiều cơ sở, đi học vất vả, tiền học rất phải suy ngẫm.
- Tên người đánh giá:** Trang ZzzZ;
Cơ sở UEH: Thư viện cơ sở N (Nguyễn Văn Linh, Bình Chánh);
Nội dung: Cơ sở vật chất tốt, máy lạnh rất lạnh. Tuy nhiên cô quản lý siêu khó tính và gắt gỏng.
- Tên người đánh giá:** Tân Nguyễn Hữu;
Cơ sở UEH: Cơ sở I (17 Phạm Ngọc Thạch, quận 3);

Nội dung: [...] lúc tuyển sinh thì nói rất hay tuy nhiên xảy ra vấn đề gì thì bị pending lại, dù tôi đã rất nhiều lần đến trường trực tiếp giải quyết, câu trả lời vẫn là “Không có quản lý đi làm, em đi về đi” và không có ngày hẹn, hay trả lời mail xác nhận nào cả. Mọi người hãy cân nhắc kỹ các vấn đề về Phòng tuyển sinh ở đây. Rất tệ về cả chất lượng lẫn tư cách làm việc của những người đã làm việc với tôi [...].

Qua một số bình luận tiêu cực, nhóm thấy tập trung các vấn đề chủ yếu liên quan tới các yếu tố như:

- Sự chuyên nghiệp và thái độ phòng ban: Nhiều người cho rằng chương trình đào tạo chưa có yếu tố chuyên môn vững vàng và một số phòng ban thiếu thái độ nhiệt huyết, thờ ơ với người học. Hoặc chất lượng đào tạo của trường được cho rằng không đáp ứng được nhu cầu học tập của các sinh viên.
- Kiến thức chuyên môn và phương pháp giảng dạy: Một vài sinh viên sau khi học có nhận xét rằng một số giảng viên có phương pháp giảng dạy khá khó hiểu và không rõ ràng, khiến sinh viên khó tiếp thu. Điều này có thể khiến sinh viên gặp khó khăn trong học tập và thi cử.

Tuy nhóm không có quá nhiều kỹ năng chuyên môn đào tạo hoặc các lĩnh vực liên quan, nhưng nhóm cũng thực hiện đề xuất một số giải pháp cải thiện như sau nhằm xây dựng môi trường học tập thân thiện:

- Nhà trường có thể tăng cường đào tạo, bồi dưỡng nâng cao kiến thức và kỹ năng cho các phòng ban mà người học gặp vấn đề. Bằng cách tổ chức các khóa đào tạo, bồi dưỡng chuyên môn và nghiệp vụ cho văn phòng, điều này sẽ giúp cập nhật kiến thức mới và giảm thiểu rủi ro mâu thuẫn với người học.
- Ngoài ra để người học có thể dễ dàng tiếp cận kiến thức hơn, văn phòng đào tạo có thể tổ chức các môn học đề cao tính thực tiễn trong chương trình giảng dạy hơn. Hoặc tổ chức những buổi tham quan doanh nghiệp theo các chuyên ngành để sinh viên có thể dễ dàng tiếp thu và áp dụng những kiến thức đã học để thực hành.

ii. **Chủ đề Cơ sở vật chất**

Đối với chủ đề này, nhóm thực hiện phân tích những bình luận tiêu cực về trang thiết bị và các cơ sở vật chất trong khuôn viên trường như bãi xe, thư viện, phòng học... Đây cũng là một chủ đề tốt để khai thác bởi vì cơ sở vật chất là vấn đề mà người học đặc biệt chú trọng quan tâm khi lựa chọn xét tuyển đầu vào.

Tương tự với phương pháp thực hiện của chủ đề đầu tiên, nhóm cũng liệt kê 10 từ khóa tiếng Việt và các từ tiếng Anh của các từ đó, đó là “*cơ sở vật chất*”, “*thiết bị*”, “*phòng học*”, “*thư viện*”, “*nhà xe*”, “*bãi xe*”, “*chỗ gửi xe*”, “*chỗ để xe*”, “*phòng lab*”, “*không gian*”. Kết quả trả về có khoảng 25 bình luận có chứa một trong các từ khóa được đề cập này. Sau đây là 3 bình luận tiêu biểu nhất nhóm chuẩn hóa và liệt kê để thực hiện việc phân tích:

- **Tên người đánh giá:** Huyen Vu;
Cơ sở UEH: Cơ sở A (59C Nguyễn Đình Chiểu, quận 3);
Nội dung: This is where I learned my economics degree. It's just so small that you can't have your bike parking, because the parking is for teachers only.
- **Tên người đánh giá:** Jennifer Louis;
Cơ sở UEH: Cơ sở E (54 Nguyễn Văn Thủ, quận 1);
Nội dung: Cơ sở nhỏ xíu, cũ. Không có thư viện, không có canteen, xung quanh bán ít đồ ăn vặt, có nhiều chỗ gửi xe cho sinh viên. Chỗ gửi xe trong trường nhỏ, ra sớm mà xe để ở trong là thôi rồi khó lấy ra, đi trễ là hết chỗ để xe. Máy tính cho giảng viên xài cũng cũ rồi, máy lạnh cũng hay hư lên hư xuống. Từ lầu 4 trở lên là ghé gỗ, không có chỗ dựa. Được cái mỗi phòng có wifi riêng cho học sinh, sinh viên, giảng viên.
- **Tên người đánh giá:** Peter;
Cơ sở UEH: UEH Boutique Hotel (232/6 Võ Thị Sáu, quận 3);
Nội dung: Room: 2/5 - The quality of the room was OK but they didn't clean the room while I stayed here for a few days. Services: 1/5 - I got a discount voucher when I paid immediately - It's OK. But check your room carefully 'cause in the day I checked out, they said I had to pay for an extra as their staff said I lost my room stuff. Food: 2/5 - They served food every morning - noodle and egg, sandwich.

Một số bình luận này phản ánh khá thực tế các nhược điểm về cơ sở vật chất của trường. Nhóm thấy các vấn đề xoay quanh những yếu tố như sau:

- Sự tăng trưởng trong nhu cầu người học: Trường đang thực hiện chuyển đổi thành đại học đa ngành nên quy mô người học cũng được phát triển theo. Các khóa người học mới sẽ với số học phí tít chỉ tăng lên, điều này dẫn đến áp lực gia tăng lên chất lượng cơ sở vật chất theo nhu cầu của sinh viên. Theo khảo sát thì ở tình hình hiện tại, số lượng trang thiết bị và không gian bãi xe ở cơ sở B không đủ đáp ứng nhu cầu của người học.
- Yếu tố chuyên môn của các nhân viên trường: Các bình luận về việc nhân viên ở các bãi giữ xe thiếu thân thiện hoặc nhân viên khách sạn không

nhật tình cho thấy trường cần nâng cao chuyên môn và tinh thần phục vụ của nhân viên để tạo thiện cảm cho người học.

Nhóm cũng đề xuất một số hướng cải thiện như sau:

- Trường cần nâng cấp cơ sở vật chất hiện có, mua sắm mới các máy móc thiết bị phù hợp với nhu cầu thực tế của người học và giảng viên. Ngoài ra, trường cũng cần nâng cao hiệu quả quản lý và vận hành cơ sở vật chất.
- Đối với các cơ sở bị thiếu phòng và trang thiết bị, trường có thể xây dựng thêm hoặc hợp tác với các bãi giữ xe và canteen bên ngoài trường để đảm bảo quyền lợi cho người học, tránh tình trạng ùn tắc giao thông vào giờ cao điểm.

iii. **Chủ đề Hoạt động sinh viên**

Ở chủ đề cuối cùng này, nhóm sẽ thực hiện phân tích những vấn đề xoay quanh sinh viên như các hoạt động người học tổ chức, ý thức và thái độ của người học. Những từ khóa tiếng Việt mà nhóm đã liệt kê liên quan tới chủ đề này là “*hoạt động*”, “*sinh viên*”, “*thái độ*”, “*câu lạc bộ*”, “*đội*”, “*nhóm*”, “*sự kiện*”, “*ý thức*”, “*người học*”, “*học sinh*”.

Kết quả trả về có 25 bình luận tiêu cực liên quan tới chủ đề này. Sau đây là 3 bình luận tiêu biểu nhất nhóm chuẩn hóa và liệt kê để thực hiện việc phân tích:

- **Tên người đánh giá:** giang nhan;
- **Cơ sở UEH:** Cơ sở A (59C Nguyễn Đình Chiểu, quận 3);
- **Nội dung:** Sinh viên hơi ồn ào bất lịch sự đề nghị trường coi lại kỷ luật hơn khi ra đường tránh ảnh hưởng đến người khác kể cả xung quanh trường.
- **Tên người đánh giá:** Nguyen Xuan Thanh;
Cơ sở UEH: Cơ sở A (59C Nguyễn Đình Chiểu, quận 3);
Nội dung: Sinh viên trường ra đường đứng làm tắc nghẽn giao thông.
- **Tên người đánh giá:** Ocean's Prince;
Cơ sở UEH: Cơ sở A (59C Nguyễn Đình Chiểu, quận 3);
Nội dung: Không gian cơ sở A khá hạn hẹp, chật chội, nhưng nhiều hoạt động diễn ra, thu hút sinh viên.

Nhóm nhận thấy rằng một số yếu tố nguyên nhân chính dẫn đến các vấn đề này có thể đến từ ý thức và thái độ của các người học, một số người học không có ý thức và văn hóa ứng xử đã gây ảnh hưởng đến môi trường học tập và sinh hoạt chung. Ngoài ra, những điều này còn gây ra những tiêu cực vượt khỏi khuôn viên khi ảnh hưởng tới những khu đô thị xung quanh cơ sở nhà trường.

Từ những nguyên nhân này kết hợp với tri thức và trải nghiệm của chính sinh viên, nhóm đề xuất một số hướng cải thiện như sau:

- Nhà trường nên tổ chức thêm các buổi kỹ năng mềm nhằm tăng cường tuyên truyền, giáo dục cho người học về ý thức và thái độ trong học tập, sinh hoạt, hoạt động trong và ngoài nhà trường. Ngoài ra, trường cũng cần có các biện pháp xử lý nghiêm minh đối với người học có hành vi vi phạm ảnh hưởng nặng.
- Nhà trường cũng có thể xây dựng và triển khai những quy chế cụ thể về quản lý các hoạt động câu lạc bộ, đội, nhóm của sinh viên, bao gồm các quy định về thời gian, nội dung và quy cách hoạt động. Đồng thời, nhà trường cũng cần tăng cường kiểm tra, giám sát việc thực hiện này.

CHƯƠNG 5: HUẤN LUYỆN DỮ LIỆU DỰA TRÊN CÁC MÔ HÌNH

1. Huấn luyện mô hình

a. Sử dụng *Naïve Bayes*

Để thực hiện xây dựng mô hình, nhóm sẽ sử dụng mô hình *Naïve Bayes* của thư viện *nlTK*. Do thư mô hình cần dữ liệu đầu vào là một tuple bao gồm các token của các câu và nhãn dán cho câu đó nên đầu tiên nhóm sẽ tiến hành chỉnh dạng dữ liệu đúng dạng của thư viện yêu cầu.

Nhóm sẽ xây dựng hàm chuyển đổi cho ra được các cặp token kèm theo giá trị True và các token trong mỗi câu sẽ đi kèm với nhãn đã được gán sẵn. Giá trị True đại diện cho sự có mặt của token trong câu, điều này đơn giản hơn là gán một tham số vector cho mỗi token. Áp dụng cho bộ dữ liệu, đã được xử lý tokenize ở bước tiền xử lý, nhóm thu được kết quả như sau.

```
[({'trường': True,
  'sạch': True,
  'đẹp': True,
  'giảng_viên': True,
  'dễ': True,
  'thương_chương': True,
  'trình': True,
  'học': True,
  'tốt': True,
  'from': True,
  'ueher': True,
  '43': True},
  'positive'),
```

Hình 18: Tuple với các cặp token đi với giá trị True và nhãn “sentiment”

Sử dụng thư viện *scikit-learn* (*sklearn*) để chia tập dữ liệu thành tập huấn luyện và tập kiểm tra. Cụ thể, đầu vào của hàm *train_test_split* là 2 Series X và y. Với X chứa văn bản sau khi đã được xử lý và làm sạch, được lưu trữ trong cột “*cleaned_text*” của dataframe “*data*”, và y là dữ liệu về nhãn của các văn bản, được lưu trữ trong cột “*sentiment*” của dataframe này.

Sử dụng hàm *train_test_split()* trong *sklearn* để tách dữ liệu thành tập huấn luyện và tập kiểm tra. Mặc định, hàm *train_test_split* sẽ chia dữ liệu đầu vào thành hai phần với tỷ lệ 80% cho tập huấn luyện và 20% cho tập kiểm tra. Ta sử dụng tham số *random_state* thiết lập giá trị ngẫu nhiên được sử dụng để xáo trộn dữ liệu trước khi chia. Bằng cách đặt giá trị này thành một số cụ thể (trong trường hợp này là 42), chúng ta có thể đảm bảo rằng cùng một chia ngẫu nhiên được tạo ra mỗi lần chạy mã. Điều này hữu ích cho tính nhất quán trong quá trình tái tạo.

Tiếp theo, nhóm sẽ sử dụng hàm *Naive Bayes* của thư viện *nlTK* để huấn luyện mô hình. Sau khi quan sát, nhóm sẽ thực hiện tính toán các tham số đánh giá bao gồm: *accuracy*, *precision*, *recall*, và *f1*.

Kết quả đánh giá trên tập train:

- Độ chính xác: 0.7574404761904762
- Độ chính xác từng lớp: 0.9071312047086041
- Độ bao phủ từng lớp: 0.7574404761904762
- F1-score: 0.7950331675726464

Đánh giá trên tập test:

- Độ chính xác: 0.7928994082840237
- Độ chính xác từng lớp: 0.901806591773147
- Độ bao phủ từng lớp: 0.7928994082840237
- F1-score: 0.8222894111488276

Sau khi đã đánh giá được mô hình, ta tạo hàm `predict_sentiment` để dự đoán kết quả positive hay negative với các bình luận. Chạy thử với ngôn ngữ tiếng Anh.

```
while True:
    text = input('> Hãy nhập câu bất kỳ: ')
    if text.lower() == 'end':
        break
    else:
        sentiment = predict_sentiment(text, 'eng')
        print(f'Kết quả dự đoán: {sentiment}\n')

> Hãy nhập câu bất kỳ: good university
Kết quả dự đoán: positive

> Hãy nhập câu bất kỳ: low infrastructure but students are nice
Kết quả dự đoán: positive
```

Hình 19: Thực hiện dự đoán ngôn ngữ tiếng Anh Naive Bayes

Nhóm thực hiện chạy thử với ngôn ngữ tiếng Việt.

```
while True:
    text = input('> Hãy nhập câu bất kỳ: ')
    if text.lower() == 'end':
        break
    else:
        sentiment = predict_sentiment(text, 'vn')
        print(f'Kết quả dự đoán: {sentiment}\n')

> Hãy nhập câu bất kỳ: cơ sở vật chất đẹp, giảng viên dạy hay
Kết quả dự đoán: positive

> Hãy nhập câu bất kỳ: cơ sở vật chất đẹp tuy nhiên nhà xe khá nhỏ, đi trễ là không có chỗ để giữ xe
Kết quả dự đoán: negative

> Hãy nhập câu bất kỳ: bảo vệ hay nhân nhỏ, khó chịu với sinh viên. Sinh viên nhiệt tình, thân thiện
Kết quả dự đoán: negative
```

Hình 20: Thực hiện dự đoán ngôn ngữ tiếng Việt Naive Bayes

b. Sử dụng MaxEnt

Ở phần này, nhóm sử dụng mô hình *MaxentClassifier* trong thư viện *nlTK*. Do thư viện cần dữ liệu đầu vào là một tuple bao gồm các token của các câu và nhãn gán cho câu đó nên đầu tiên nhóm sẽ tiến hành chỉnh dạng dữ liệu đúng dạng của thư viện yêu cầu.

Nhóm sẽ xây dựng hàm chuyển đổi cho ra được các cặp token kèm theo giá trị True và các token trong mỗi câu sẽ đi kèm với nhãn đã được gán sẵn. Giá trị

True đại diện cho sự có mặt của token trong câu, điều này đơn giản hơn là gán một tham số vectơ cho mỗi token. Áp dụng cho bộ dữ liệu, đã được xử lý tokenize ở bước tiền xử lý, nhóm thu được kết quả như sau.

```
[({'trường': True,
  'sạch': True,
  'đẹp': True,
  'giảng_viên': True,
  'dễ': True,
  'thương chương': True,
  'trình': True,
  'học': True,
  'tốt': True,
  'from': True,
  'ueher': True,
  '43': True},
  'positive'),
```

Hình 21: Tuple với các cặp token đi với giá trị True và nhãn “sentiment”

Sử dụng bộ chuyển đổi trong thư viện *scikit-learn* để chia bộ dữ liệu thành 2 phần tập train và tập test với tỷ lệ 80:22 và mức *random_state* là 42 để đảm bảo tính tái tạo lại với mỗi lần chạy mới. Tiếp theo, nhóm sẽ sử dụng hàm *MaxentClassifier* của thư viện *nlTK* để huấn luyện mô hình. Cài đặt các tham số cho mô hình bao gồm:

- *trace = 3*: In ra các thông tin tiến trình huấn luyện mô hình qua mỗi vòng lặp (cụ thể ở đây là số vòng lặp, giá trị *log_likelihood* và tham số đánh giá accuracy);
- *algorithm = 'iis'*: Chọn phương pháp tối ưu hóa trong quá trình huấn luyện là Improved Iterative Scaling (đây là phương pháp tối ưu hóa để điều chỉnh trọng số *log_likelihood* cho mô hình, thích hợp cho các bộ dữ liệu có kích thước vừa và nhỏ);
- *max_iter = 20*: cài đặt số lần lặp tối đa (số epochs) mà thuật toán huấn luyện sẽ chạy. Nó xác định số lần mà thuật toán tối ưu hóa điều chỉnh tham số mô hình dựa trên dữ liệu huấn luyện để cải thiện hiệu suất. Ở đây nhóm sẽ cài đặt 20 vòng lặp cho thuật toán.

Iteration	Log Likelihood	Accuracy
1	-0.69315	0.866
2	-0.34073	0.900
3	-0.27419	0.943
4	-0.24059	0.957
5	-0.21844	0.967
6	-0.20195	0.970
7	-0.18884	0.976
8	-0.17797	0.978
9	-0.16871	0.982
10	-0.16068	0.982
11	-0.15362	0.982
12	-0.14734	0.984
13	-0.14170	0.987
14	-0.13660	0.991
15	-0.13195	0.993
16	-0.12770	0.993
17	-0.12379	0.993
18	-0.12017	0.993
19	-0.11681	0.993
Final	-0.11369	0.993

Hình 22: Vòng lặp đạt kết quả tối ưu

Sau khi quan sát nhóm nhận thấy kết quả tối ưu dựa trên tham số đánh giá accuracy là 0.993 ở vòng lặp thứ 15, các vòng lặp tiếp theo không mang lại sự tăng nhiều cho giá trị tham số accuracy. Kể từ đây để đánh giá mô hình nhóm sẽ thực hiện tính toán các tham số đánh giá như accuracy, precision, recall, và f1-score.

Bảng đánh giá

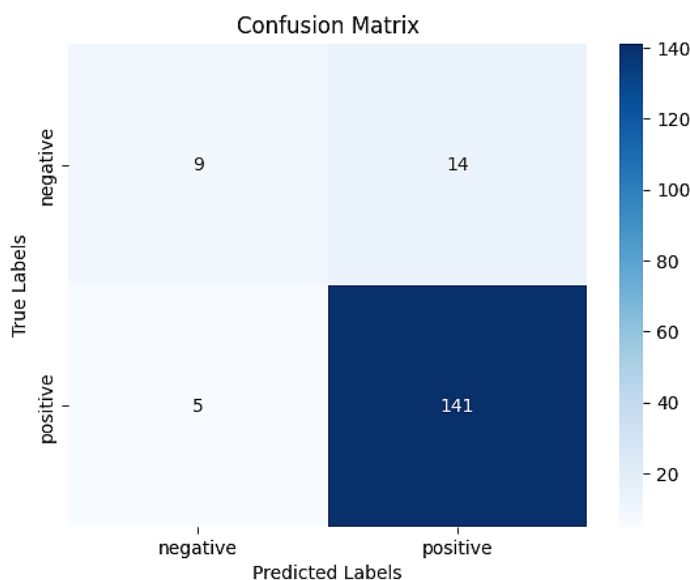
Metrics	P_Train	P_Test	N_train	N_test
Accuracy	0.9925595238095238	0.8875739644970414	0.9925595238095238	0.8875739644970414
Precision	0.9926034593633561	0.873364600659886	0.967032967032967	0.6428571428571429
recall	0.9925595238095238	0.8875739644970414	0.9777777777777777	0.391304347826087
f1	0.9925768787821823	0.875581315652872	0.9723756906077348	0.4864864864864865

Hình 23: Bảng kết quả đánh giá

Kết quả đánh giá cho thấy trên tập train, mô hình đạt điểm số gần như tối đa khi cả bốn chỉ số đều đạt gần 100% đồng nghĩa với việc gần như không một dòng review nào bị đánh giá sai. Thậm chí đối với điểm precision của nhãn “negative” đạt điểm gần tối đa, khoảng lớn hơn 96%, tức là có rất ít dòng có nhãn ‘positive’ nào bị gán nhãn nhầm lẫn.

Tuy nhiên, khi tham chiếu trên tập test, các tham số đánh giá bị giảm khá mạnh cụ thể là ở các dòng có nhãn “negative”, thấp nhất là giá trị điểm recall của nhãn này chỉ đạt 39.13%, trong khi trên tập huấn luyện đây là giá trị đạt điểm số rất cao. Điều này cho thấy mô hình có khả năng dự đoán nhầm lẫn rất cao ở đối với các dòng đánh giá tiêu cực ở tập test.

Có thể giải thích vấn đề này do ảnh hưởng của bộ dữ liệu nhóm sử dụng không có được sự cân bằng giữa 2 nhãn “positive” và “negative” dẫn đến mô hình đã có xu hướng gán nhãn tích cực cho các dòng giá trị mới. Để có thể quan sát kết quả kiểm tra trực quan, nhóm sẽ tiến hành vẽ biểu đồ ma trận nhầm lẫn cho các giá trị được gán nhãn trong tập kiểm tra.



Hình 24: Ma trận nhầm lẫn cho tập test

Ma trận nhầm lẫn cho thấy số dòng được dự đoán nhãn “negative” là 15 dòng và có đến 33.33% trong số này bị dự đoán sai. Trong khi đó, 141 dòng “positive” đã được dự đoán đúng trên tổng số 146 dòng được dự đoán, điều này cho thấy mô hình hoạt động gán nhãn cho các dòng có xu hướng tích cực là khá tốt nhưng lại vô cùng tệ khi làm việc với các dòng có xu hướng tiêu cực. Để minh họa cụ thể hơn, nhóm sẽ tiến hành thực hiện gán nhãn cho một số câu review nằm ngoài bộ dữ liệu.

Predicted Sentiment for 'Giảng viên tệ, không hết mình': negative
 Predicted Sentiment for 'Sinh viên ồn ào, bãi xe chật chội': negative
 Predicted Sentiment for 'One of the best university in VietNam': positive
 Predicted Sentiment for 'Trường học khang trang mát mẻ, chất lượng đào tạo tốt': positive
 Predicted Sentiment for 'Chất lượng giảng đường không tốt, không khí ngột ngạt': positive

Hình 25: Kết quả thực nghiệm dự đoán cho 5 câu đánh giá bất kỳ

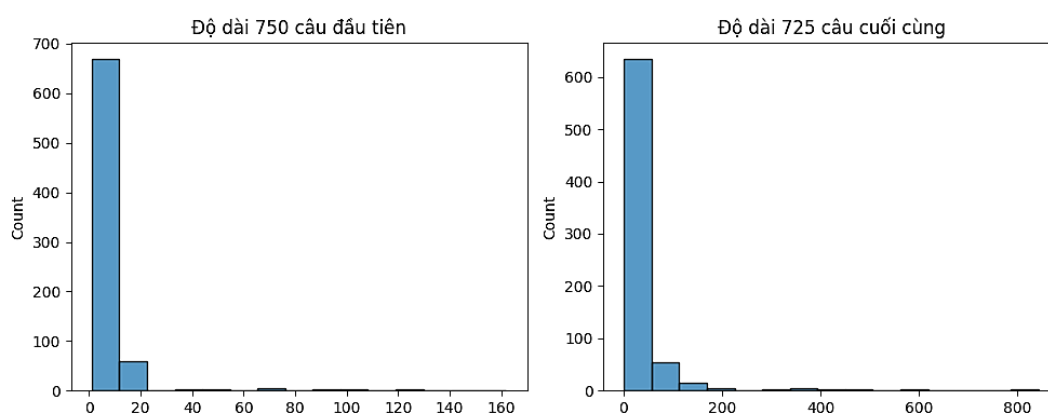
Có thể thấy, mô hình đã dự đoán đúng 4 trên 5 câu đánh giá ngẫu nhiên mà nhóm đã chọn lựa. Những câu bình luận được gán nhãn ‘negative’ là những câu được nhóm chọn lựa từ ngữ gần với các từ có trong những review tiêu cực trong bộ dữ liệu thì mô hình cho thấy khả năng dự đoán rất tốt, tuy nhiên đối với câu thứ 5, nhóm đã cố ý chọn các từ ngữ có phần nghi ngờ về các câu đánh giá tích cực trong bộ dữ liệu thì mô hình đã dự đoán sai về trường hợp này.

c. Sử dụng CNN và BiLSTM

Ở phần này, nhóm thực hiện sử dụng hai mô hình học sâu thông dụng để thực hiện huấn luyện mô hình là CNN và BiLSTM với framework chính từ thư viện *keras*.

Tại bước chuẩn bị dữ liệu, nhóm đã tham khảo nhiều phương pháp thực hiện khác nhau và cuối cùng quyết định tăng số lượng quan sát bộ dữ liệu bằng cách nhân bản số lượng quan sát thành 2 loại: loại có dấu như cơ bản và loại không có dấu câu.

Sau khi thực hiện, nhóm kiểm tra và thấy số lượng quan sát lúc này khoảng 1682 dòng, vậy nên nhóm sẽ chia bộ dữ liệu này ra làm 2 khoảng và thực hiện trực quan hóa để kiểm tra xem độ dài của các câu tập trung nhiều ở những khoảng nào để quyết định chọn độ dài tối đa của vector ở khoảng đó. Lý giải cho điều này bởi vì nếu như nhóm chọn độ dài của vector quá lớn, dữ liệu sẽ gặp tình trạng bị thừa và tốn quá nhiều tài nguyên, còn vector có độ dài quá nhỏ thì sẽ không thể biểu diễn được hết tất cả ý nghĩa của câu được truyền vào.



Hình 26: Biểu đồ Histogram thể hiện độ dài của các câu

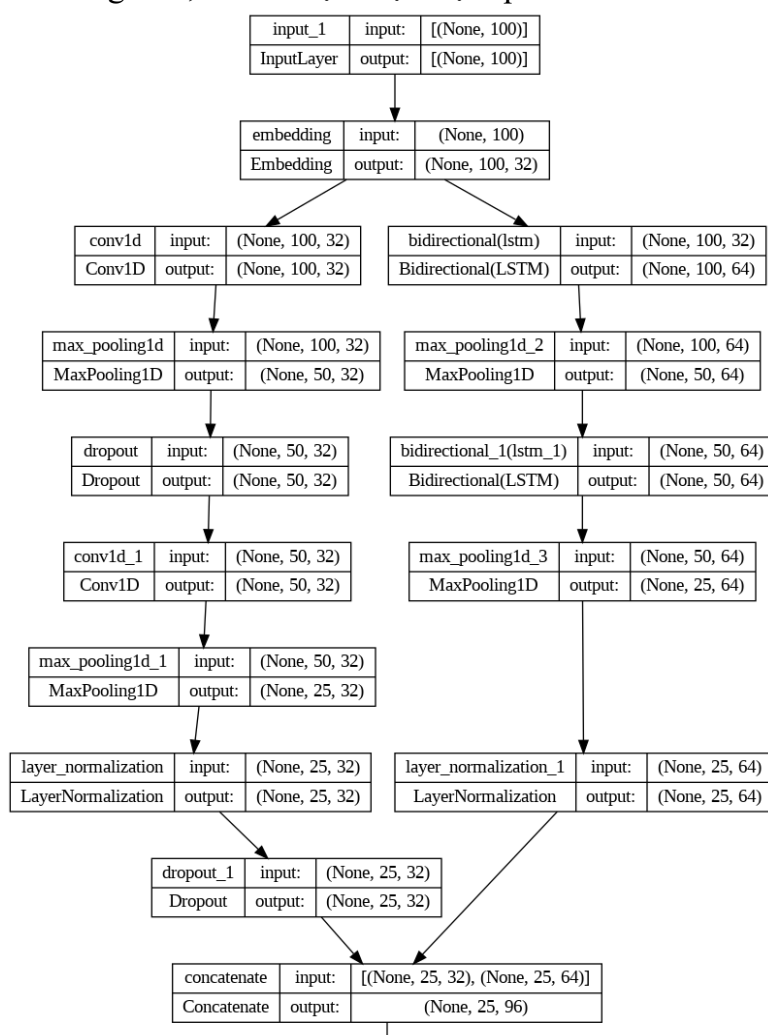
Qua hai biểu đồ trực quan, nhóm nhận thấy rằng đa số dữ liệu có chiều dài câu tập trung nhiều ở khoảng 600, vậy nhóm sẽ sử dụng vector có độ dài tối đa là 600 và chấp nhận đánh đổi mất mát thông tin đối với những câu có số từ dài hơn mức này.

Tiếp theo, nhóm thực hiện chuẩn bị và phân chia tập dữ liệu cho mô hình học máy. Đầu tiên, nhóm thực hiện quá trình vector hóa văn bản (text vectorization) bằng *Tokenizer*, ở đây với tập dữ liệu ban đầu khá hạn hẹp về khía cạnh từ vựng, nhóm sử dụng siêu tham số *oov_token* để thực hiện thay thế các từ không được xuất hiện trong tập dữ liệu huấn luyện, những từ này sẽ được thay thế bằng “<OOV>”, điều này giúp cải thiện sự hiệu quả của mô hình.

Ngoài ra, đối với việc bảo đảm chuỗi số nguyên này có đầu ra chiều dài cố định, giúp các chuỗi về cùng một kích thước để tạo ma trận đầu vào có shape đồng

nhất. Ở đây, nhóm đã sử dụng phương pháp *pad_sequences* với siêu tham số *padding* là “*post*”, tham số này chủ yếu thực hiện thay đổi ở đoạn phía sau của chuỗi khi nó sẽ cắt bớt phần dư nếu chuỗi dài hơn thông số *maxlen* chỉ định, nếu chuỗi ngắn hơn, nó sẽ được thêm vào các giá trị để đạt đến ngưỡng chỉ định (thường là giá trị 0). Kết quả của quá trình này sẽ tạo thành một ma trận đầu vào với kích thước xác định.

Và cũng để đảm bảo nhóm có thể tái sử dụng quy trình này cho dữ liệu mới mà không cần phải thực hiện lại quá trình xây dựng như các bước vừa rồi, nhóm đã lưu lại *Tokenizer* thành “*tokenizer_data.pkl*” để tiếp tục sử dụng *Tokenizer* này huấn luyện mô hình khi cần thiết. Cuối cùng, nhóm thực hiện phân chia bộ dữ liệu sau khi xử lý này thành các tập train, validation, test theo tỷ lệ 80:20 để kiểm thử và đánh giá hiệu suất của mô hình trước và sau khi huấn luyện. Sau khi hoàn thành bước chuẩn bị dữ liệu, nhóm tiến hành xây dựng mô hình học sâu kết hợp và để dễ hình dung hơn, nhóm thực hiện trực quan hóa như sau.



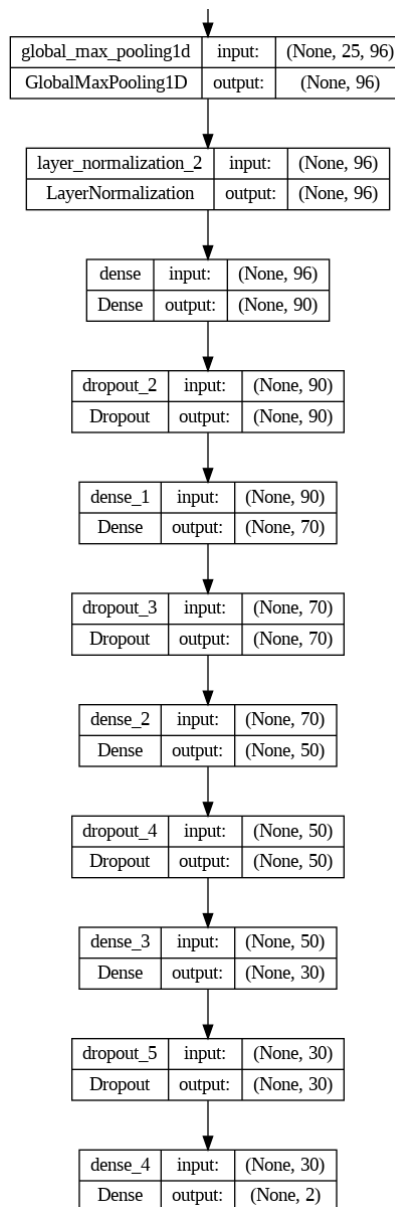
Hình 27: Kiến trúc model xây dựng (từ bước input tới bước kết hợp mô hình)

Đầu tiên, nhóm bắt đầu bằng việc khởi tạo các siêu tham số trong mô hình: initializer là phương pháp khởi tạo trọng số của các layer, ở đây nhóm sử dụng *GlorotNormal()* trong thư viện *keras* giúp giảm biến động của trọng số được khởi tạo trong quá trình huấn luyện. Ngoài ra, nhóm cũng có thể dùng một số phương pháp khác như *RandomUniform()* hoặc *HeNormal()*.

Tiếp theo, nhóm tạo Embedding layer để chuyển đổi các từ trong văn bản thành các vector có kích thước cố định là 32. Điều này giúp biểu diễn các từ dưới dạng các điểm trong không gian vector, làm mô hình học được sự tương đồng về ý nghĩa giữa các từ. Nếu không sử dụng Embedding Layer, mô hình sẽ phải sử dụng *one-hot encoding* cho mỗi từ để có thể chuyển đổi thành vector, nhưng đối với ngữ cảnh bài toán sử dụng các câu có độ dài lớn như hiện tại thì tính khả thi của việc này là không thể.

Sau khi khởi tạo thành công layer Embedding, nhóm tiếp tục chia mô hình thành 2 hướng: một hướng sẽ đi qua mô hình CNN và một hướng đi qua mô hình BiLSTM. Sau đó, nhóm tổng hợp thông tin hai mô hình này học được, bằng cách này, mô hình có thể vừa hiểu được đặc trưng từ mô hình CNN như feature mapping và feature vectors, vừa học được mối quan hệ giữa các từ trong câu theo mô hình BiLSTM. Lưu ý rằng, để có thể kết hợp được 2 mô hình vào cấu trúc model tổng quát, đầu ra của 2 mô hình này cần tương đương nhau.

Tiếp theo, nhóm thực hiện xây dựng các Dense layer để thêm các lớp fully connected nhằm giúp mô hình học được các mối quan hệ trong dữ liệu. Các siêu tham số như *activation*, *units* hay *dropout* được điều chỉnh để đảm bảo tính linh hoạt và tránh *overfitting*.



Hình 28: Kiến trúc model xây dựng (từ bước kết hợp mô hình tới bước output)

Bài toán được xác định là phân lớp với 2 nhãn chính là “positive” và “negative”, vậy nên ở layer cuối cùng, nhóm sử dụng hàm kích hoạt (activation function) là hàm “softmax”. Nhóm cũng thực hiện trực quan cấu trúc này để hiểu rõ hơn về model.

Sau khi tạo thành công được model, nhóm sử dụng hàm callback *ModelCheckpoint()* để thực hiện chuẩn bị lưu trữ model sau mỗi epoch nếu giá trị loss trên tập validation giảm, từ đó, lưu lại mô hình có hiệu suất tốt nhất.

Sau đó, nhóm thực hiện train mô hình với số lần là 20 epochs và kích thước batch là 128, mỗi epoch đại diện một lần duyệt qua toàn bộ tập train và mỗi lần cập nhật trọng số được thực hiện với các batch có kích thước 128. Cuối cùng sau khi train, nhóm nhận được một mô hình và đặt tên là “CNN-BILSTM_model.h5”.

	precision	recall	f1-score	support
0	0.62	0.66	0.64	32
1	0.95	0.94	0.94	204
accuracy			0.90	236
macro avg	0.78	0.80	0.79	236
weighted avg	0.90	0.90	0.90	236

Hình 29: Kết quả classification report từ mô hình

Dựa trên matrix, nhóm nhận thấy có một số thông tin như sau:

- Tuy mô hình có độ chính xác khá cao (khoảng 90%) nhưng các chỉ số recall và f1-score của phần tử nhãn 0 (tương ứng với “negative”) là khá thấp.
- Nhóm nhận thấy weighted average cao hơn rất nhiều so với macro average chứng tỏ có một nhãn có số lượng mẫu lớn hơn (tỷ lệ recall chênh lệch 10%).

Vậy qua báo cáo phân lớp, nhóm nhận thấy rằng hai label đang có dấu hiệu chênh lệch khá lớn, điều này có thể làm mất cân bằng dữ liệu và khiến hiệu suất dự đoán mô hình giảm đi rõ rệt. Qua nhiều cách cải tiến tham khảo, nhóm quyết định sử dụng một phương pháp cơ bản nhưng hiệu quả, đó là sử dụng parameter *class_weight* trong hàm *fit()* để thực hiện gán trọng số weight cho từng label.

	precision	recall	f1-score	support
0	0.63	0.69	0.66	32
1	0.95	0.94	0.94	204
accuracy			0.90	236
macro avg	0.79	0.81	0.80	236
weighted avg	0.91	0.90	0.90	236

Hình 30: Kết quả classification report từ mô hình gán trọng số nhãn

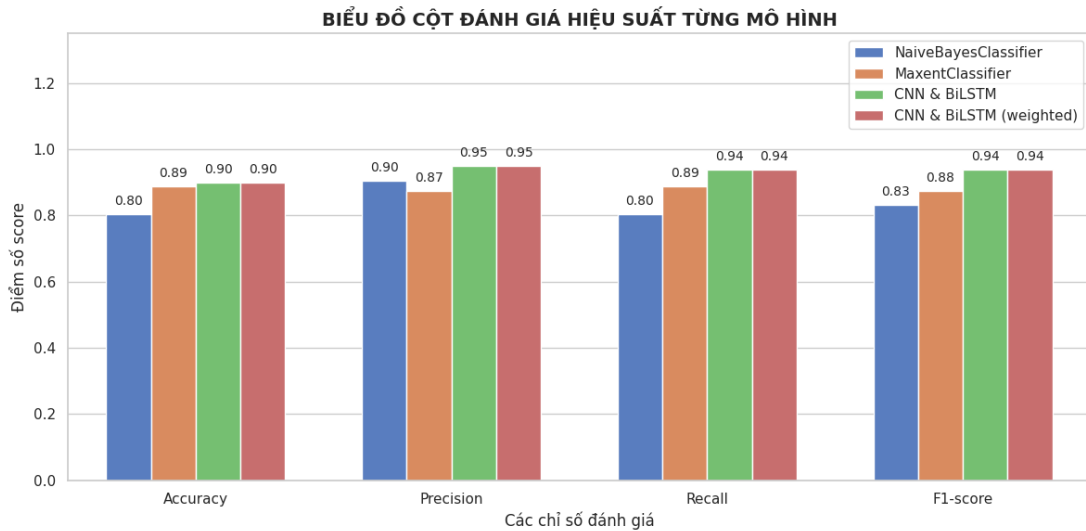
Nhóm thấy được sau khi cải thiện bằng phương pháp gán trọng số theo nhãn, độ chính xác tổng thể của mô hình đã giảm xuống, đặc biệt với nhãn “negative”. Tuy nhiên, chỉ số recall có sự tăng mạnh chứng tỏ mô hình hiệu quả hơn trong việc dự đoán chính xác các trường hợp đúng (true positive).

Việc này cũng có nghĩa là nhóm sẽ phải đánh đổi giữa các chỉ số khác như precision, điều này sẽ làm các trường hợp rơi vào diện false positive có khả năng cao hơn so với mô hình cũ. Tuy nhiên trong thực tế, ban đầu nhóm đã sử dụng mô hình đầu tiên để thực hiện dự đoán thử và thấy được rằng các dự đoán của mô hình này hoàn toàn lệch về nhãn 1 (tương ứng với bình luận tích cực) đối với

cả 5 câu training có nhãn là negative. Mặc dù accuracy rất cao nhưng đối với bộ dữ liệu imbalance, nhóm quyết định sử dụng mô hình đề cao chỉ số recall hơn.

2. Đánh giá so sánh trực quan

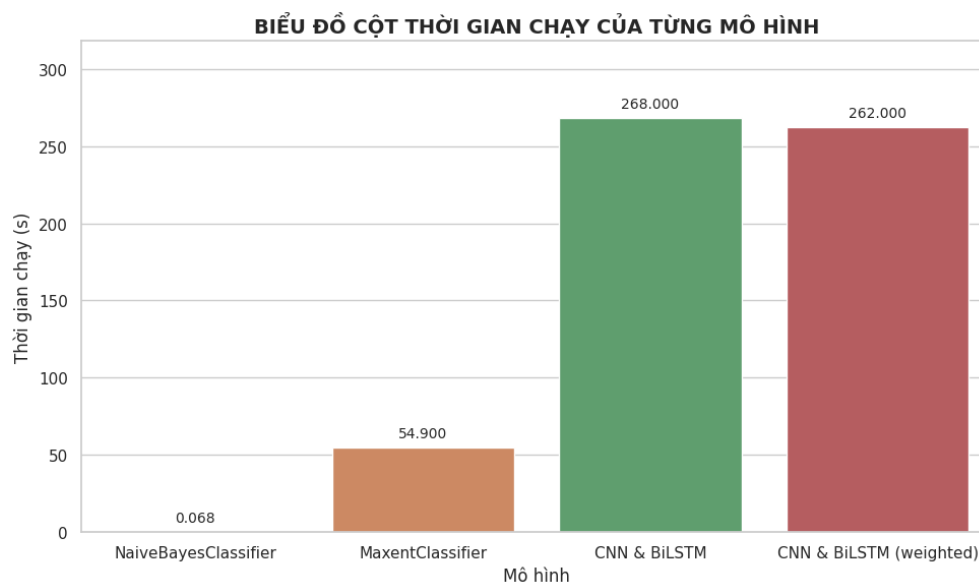
Sau khi xem xét cả ba mô hình dựa trên các chỉ số đánh giá, nhóm thực hiện trực quan kết quả để thuận tiện cho việc so sánh.



Hình 31: Biểu đồ cột cụm thể hiện các chỉ số đánh giá qua từng mô hình

Nhóm nhận thấy rằng nhóm mô hình học sâu (mô hình sử dụng CNN kết hợp BiLSTM trước và sau khi sử dụng trọng số nhãn) có chỉ số tốt và ổn định nhất với các mức đều trên 90%, điều này chứng tỏ khả năng phân lớp dự đoán của hai loại mô hình này khá tốt đối với bộ dữ liệu của bài toán này. Ngoài ra, các mô hình khác cũng cho mức chỉ số ở mức khá tốt lần lượt là trên 80% (đối với Naïve Bayes) và trên 87% (đối với Maxent).

Tuy nhiên khi xem xét kỹ hơn, mô hình học sâu sử dụng trọng số nhãn có chỉ số recall nhãn “negative” cao hơn so với khi không sử dụng. Vấn đề này phát sinh do bộ dữ liệu có số lượng quan sát thuộc về một nhãn quá lệch so với nhãn còn lại, chi tiết hơn recall đã tăng từ 0.66 lên 0.69, ngoài ra đối với các lần thử nghiệm trước, nhóm đã từng thấy recall của mô hình không sử dụng trọng số nhãn giảm chỉ còn 0.3 và mô hình sử dụng trọng số tăng đến mức 0.8, điều này cho thấy các lần thực nghiệm khác nhau với tập dữ liệu huấn luyện và kiểm thử giống nhau, mô hình gán trọng số sẽ thường cho kết quả tốt hơn mô hình không gán trọng số.



Hình 32: Biểu đồ cột thể hiện thời gian huấn luyện mô hình của từng mô hình

Nhóm sử dụng extension của ipython để xem xét thời gian chạy của từng cell, đó là “**%load_ext time**”. Sau khi thực hiện, nhóm nhận thấy thời gian huấn luyện của từng mô hình lần lượt là 0.068 giây (NaïveBayesClassifier), 54.9 giây (MaxentClassifier), 268 giây (CNN & BiLSTM) và 262 giây (CNN & BiLSTM sử dụng trọng số nhân). Tuy nhiên cũng cần lưu ý rằng thời gian chạy các mô hình này phụ thuộc khá nhiều vào môi trường chạy mã nguồn, ở đây nhóm sử dụng môi trường trực tuyến Colab và cho thời gian như trên.

Vậy nhóm có thể thấy được hai mô hình học sâu có thời gian chạy lâu nhất, còn mô hình Naïve Bayes sẽ có thời gian chạy nhanh nhất, nhanh hơn hẳn so với các mô hình còn lại, chứng tỏ mô hình này hợp lý để sử dụng khi không cần đặt nặng về tỷ lệ dự đoán quá nhiều. Còn các mô hình học sâu do phải huấn luyện qua nhiều layer nên thời gian sẽ lâu hơn nhưng đánh đổi lại, tỷ lệ đoán chính xác khá cao nếu bài toán đề cao tỷ lệ dự đoán chính xác.

3. Ứng dụng kết quả (Demo)

Ở phần này, nhóm thực hiện triển khai mô hình thu được để xây dựng ứng dụng từ kết quả này. Qua nhiều tài liệu tham khảo, nhóm quyết định chọn phương pháp sử dụng Flask, một framework sử dụng ngôn ngữ Python, kết hợp với *html* để xây dựng trang web ứng dụng.

```

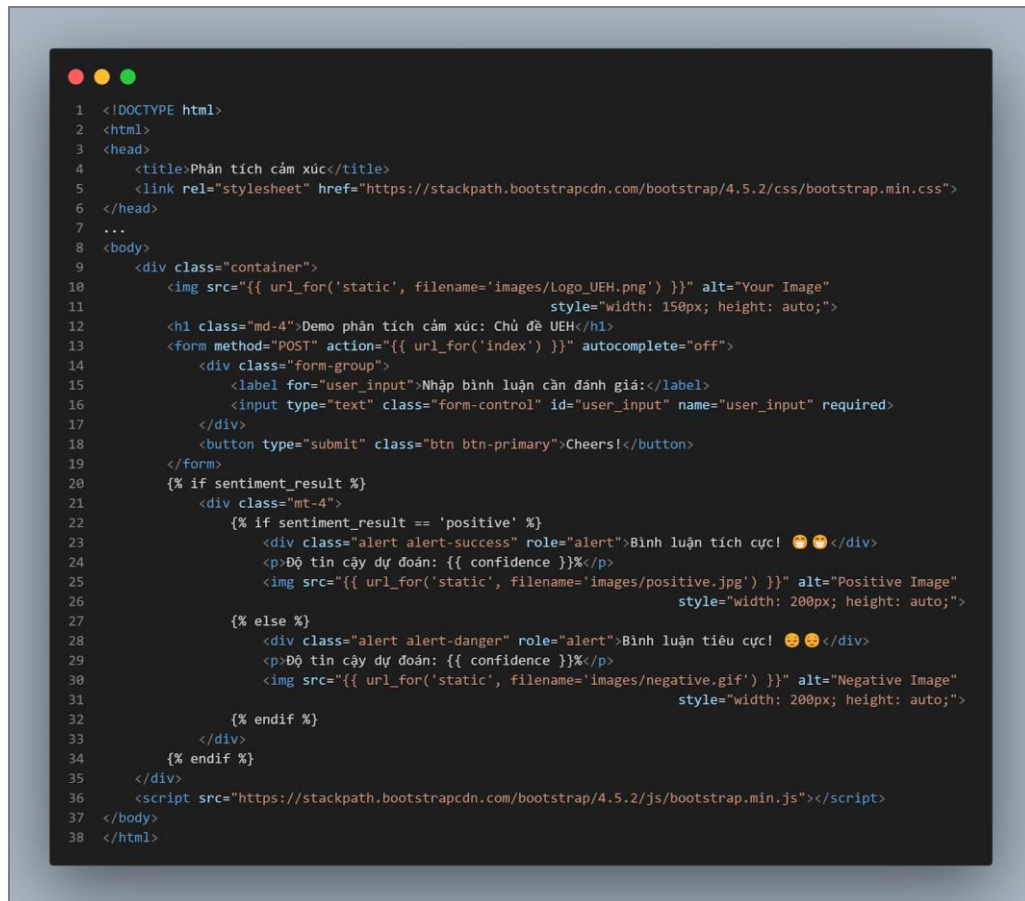
1 from flask import Flask, render_template, request
2 import tensorflow as tf
3 import pickle as pkl
4 from pyvi import ViTokenizer
5 from tensorflow.keras.models import load_model
6 from tensorflow.keras.preprocessing.sequence import pad_sequences
7
8 app = Flask(__name__)
9 with open('tokenizer_data.pkl', 'rb') as file:
10     my_tokenizer = pkl.load(file)
11 my_model = load_model('CNN-BILSTM_model.h5', compile=False)
12 maxlen_vector = 600
13
14 def get_vectorize(input, tokenizer):
15     input_text_pre = list(tf.keras.preprocessing.text.text_to_word_sequence(input))
16     input_text_pre = ' '.join(input_text_pre)
17     input_text_pre_accent = ViTokenizer.tokenize(input_text_pre)
18     tokenized_data_text = tokenizer.texts_to_sequences([input_text_pre_accent])
19     vec_data = pad_sequences(tokenized_data_text, padding='post', maxlen=600)
20     return vec_data
21
22 def get_confidence(feature, model):
23     label_dict = {'negative': 0, 'positive': 1}
24     label = list(label_dict.keys())
25     output = model(feature).numpy()[0]
26     result = output.argmax()
27     conf = round(float(output.max()), 4) * 100
28     return label[int(result)], conf
29
30 def get_prediction(input, tokenizer, model):
31     input_model = get_vectorize(input, tokenizer)
32     result, conf = get_confidence(input_model, model)
33     return result, conf
34
35 @app.route('/', methods=['GET', 'POST'])
36 def index():
37     sentiment_result = None
38     confidence = None
39     if request.method == 'POST':
40         user_input = request.form['user_input']
41         if user_input:
42             sentiment_result, confidence = get_prediction(user_input, my_tokenizer,
43                                                         my_model)
44     return render_template('home.html', sentiment_result=sentiment_result,
45                           confidence=(confidence))
46
47 if __name__ == '__main__':
48     app.run(debug=True)

```

Hình 33: Ứng dụng Flask để xây dựng trên mô hình học sâu

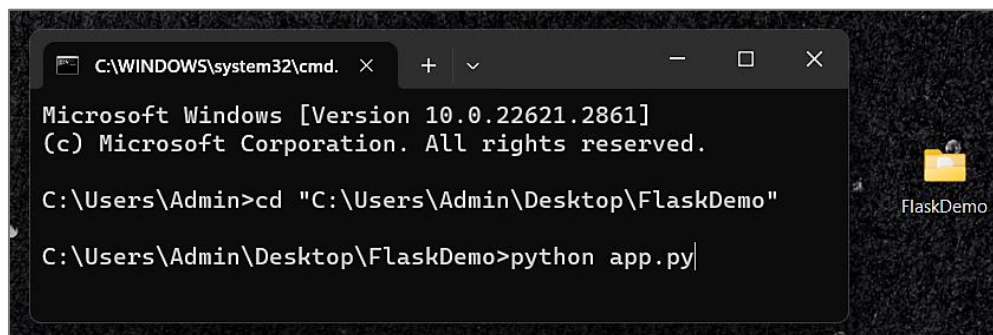
Các hàm nhóm sử dụng ở đây tương tự với những hàm lúc thực hiện dự đoán ở phần mô hình học sâu, các file *model* và *tokenizer* đã được lưu từ công đoạn *fit* mô hình sẽ được nhóm sử dụng lại để nhằm tiết kiệm thời gian huấn luyện. Nhóm đặt tên cho tập tin này là “*app.py*”.

Ngoài ra để trang web thân thiện với người dùng hơn, nhóm cũng thực hiện xây dựng thêm trong tập tin *.html* (nhóm đặt tên là “*home.html*”), chi tiết đoạn mã như sau.



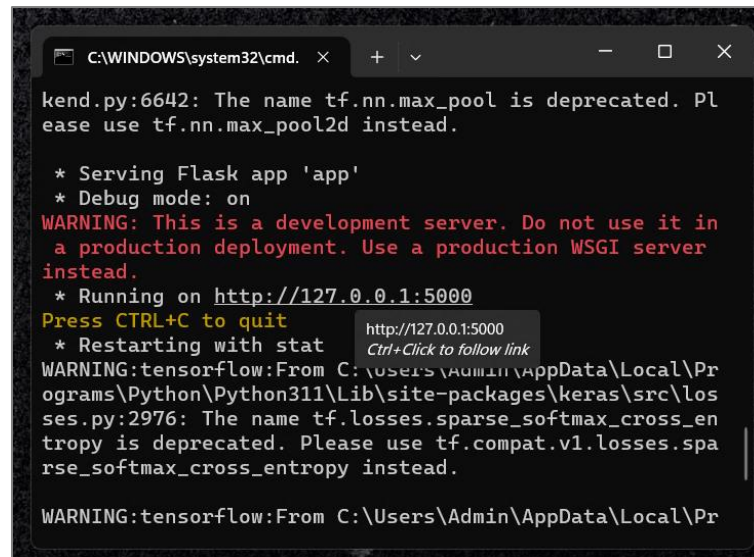
Hình 34: Xây dựng giao diện web bằng html:5

Vì chưa có kinh nghiệm lập trình HTML/CSS, nhóm đã tham khảo nhiều tài liệu khác nhau và đây là chi tiết giao diện cuối cùng mà nhóm xây dựng, phần “...” chỉ là đoạn mã mà nhóm trang trí thêm nên nhóm sẽ tạm lược bớt trong báo cáo.



Hình 35: Giới thiệu cách khởi chạy demo

Để khởi chạy ứng dụng vừa xây dựng được, ta cần thực hiện truy cập terminal bằng cmd vào thư mục đang chứa tập tin xây dựng bằng Flask và khởi chạy nó. Lưu ý để thực hiện chạy được file python, ta cần cài đặt các thư viện nhóm sử dụng trong tập tin này, các thư viện này bao gồm: *flask*, *tensorflow*, *keras*, *pickle* và *pyvi*. Ta có thể sử dụng hàm “**python -m pip install <thư viện>**” để cài đặt những thư viện đó trong terminal.



```
C:\WINDOWS\system32\cmd. x + v - □ x

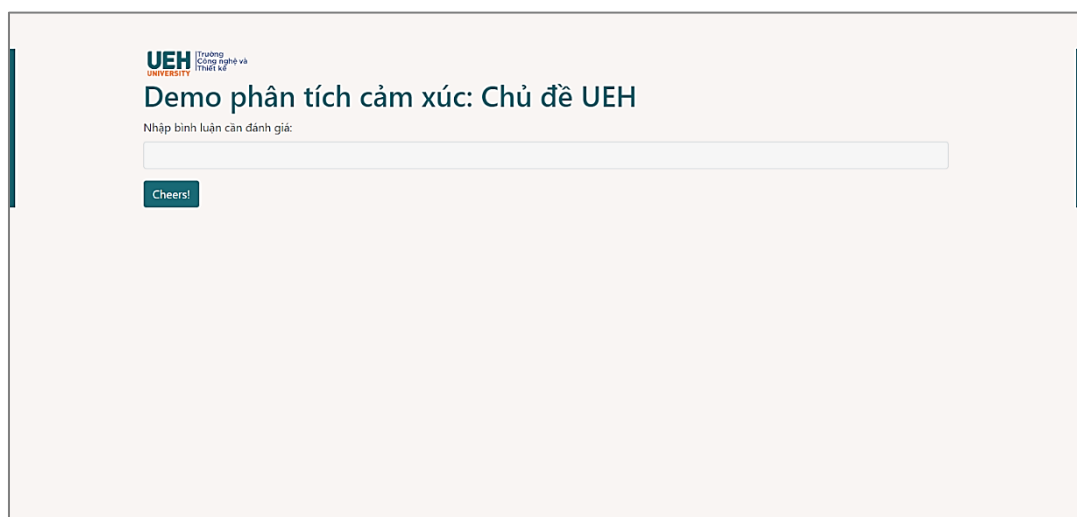
kend.py:6642: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.

* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat http://127.0.0.1:5000
Ctrl+Click to follow link
WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Programs\Python\Python311\Lib\site-packages\keras\src\losses.py:2976: The name tf.losses.sparse_softmax_cross_entropy is deprecated. Please use tf.compat.v1.losses.sparse_softmax_cross_entropy instead.

WARNING:tensorflow:From C:\Users\Admin\AppData\Local\Pr
```

Hình 36: Khởi chạy development server xây dựng bằng Flask

Sau khi mở đường dẫn bằng browser, ta có thể nhập vào bình luận bất kỳ để thực hiện phân lớp kiểm tra xem câu bình luận đó là tích cực (positive) hoặc tiêu cực (negative).



Hình 37: Website demo lúc vừa khởi chạy development server

Nhóm tiến hành nhập một nhận xét bất kỳ vào ô bình luận và nhấn nút phân tích. Ở đây nhóm sẽ thử nghiệm với câu “Các thầy cô rất nhiệt tình giảng dạy sinh viên.”, kết quả trả về câu bình luận này dự đoán phân vào nhãn tích cực (positive) với độ chính xác tính được theo mô hình học sâu mà nhóm đã thực hiện.



Hình 38: Website demo sau khi thực nghiệm với bình luận bất kỳ

CHƯƠNG 6: KẾT LUẬN VÀ ĐÁNH GIÁ

1. Kết quả đạt được

Thông qua quá trình tìm hiểu và nghiên cứu, nhóm đã xây dựng được mô hình phân lớp phân tích phản hồi của khách hàng về Đại học UEH (tích cực/tiêu cực) bằng mô hình Naïve Bayes, Maxent, CNN & BiLSTM và kết quả cuối cùng cho thấy mô hình học sâu kết hợp CNN và BiLSTM sử dụng trọng số cho kết quả dự đoán tốt nhất, tiếp theo là Maxent và Naïve Bayes. Đồng thời nhóm đã ứng dụng những kiến thức đã học về Xử lý ngôn ngữ tự nhiên để thực hiện khai thác và xử lý dữ liệu văn bản.

2. Hạn chế

Nhóm nhận thấy qua thực nghiệm, nhóm chưa khai thác và sử dụng được hết toàn bộ dữ liệu như các thông tin về cơ sở của trường hoặc các khía cạnh khác của trường trong việc phân lớp đánh giá người dùng để có thể phân tích nguyên nhân của các đánh giá tiêu cực đồng thời đưa ra hướng cải thiện. Bộ dữ liệu ban đầu sau khi cào khá ít (khoảng 900 quan sát), việc thêm dữ liệu các quan sát sẽ cải thiện việc xây dựng mô hình phân lớp tốt hơn, tránh tình trạng dữ liệu bị mất cân đối. Ngoài ra, phần đánh giá sử dụng khá nhiều teencode, và lỗi chính tả của các nhận xét cũng khá nhiều nên nhóm không thể xử lý hoàn toàn, làm ảnh hưởng đến hiệu quả xây dựng mô hình.

3. Các phương pháp cải tiến đề xuất

Nhóm đề xuất một số phương pháp cải tiến như sau:

- **Cải thiện việc tiền xử lý và làm sạch dữ liệu:** Nhóm cần cải thiện quá trình xử lý các đánh giá này tốt hơn để lọc ra những bình luận không phù hợp như các nhận xét toxic, không liên quan. Ngoài ra, nhóm cũng cần xử lý các đánh giá bị sai lỗi chính tả tiếng Việt, từ teencode và các từ bị đánh máy sai.
- **Sử dụng *focal loss* để giảm độ mất cân bằng của bộ dữ liệu:** Nhóm được đề xuất sử dụng “*focal loss*” thay cho “*binary crossentropy*” do bộ dữ liệu bị mất cân bằng giữa 2 nhãn “*sentiment*”. Cơ chế chính của *focal loss* là sử dụng tham số gamma (một hằng số dương) để giảm trọng số của các mẫu dễ phân loại đúng (ở đây là nhãn “*positive*”), khi giá trị gamma tăng lên, mức độ giảm trọng số càng lớn, đồng thời mô hình tập trung nhiều hơn vào việc học từ các mẫu khó.

Lý do nhóm phải sử dụng phương pháp này vì đây là phương án tối ưu nhất do bộ dữ liệu không những bị mất cân bằng giữa 2 nhãn phân loại mà còn có kích thước không lớn, đặc biệt là nhãn “*negative*” chỉ có 127 quan sát nên không thể áp dụng các phương pháp xử lý khác như *random undersampling*. Phương pháp này sẽ giảm kích thước theo nhãn có lượng mẫu nhỏ hơn, vậy nếu áp dụng phương pháp này bộ dữ liệu huấn luyện sẽ bị giảm còn khoảng 254 quan sát, và thế là tập huấn luyện sẽ không đủ lớn để hoạt động hiệu quả các mô hình học sâu.

DANH MỤC HÌNH ẢNH

Hình 1: Mô phỏng quy trình xử lý ngôn ngữ tự nhiên dựa trên mô hình CNN (theo Dennybritz).....	5
Hình 2: Lưu đồ quy trình xử lý dựa trên mô hình BiLSTM (theo Baeldung).....	6
Hình 3: Giao diện tổng quan trích xuất dữ liệu bằng phần mềm Octoparse	7
Hình 4: Cây quy trình trích xuất dữ liệu thực hiện bằng Octoparse	8
Hình 5: Giao diện kết quả trích xuất dữ liệu với địa điểm “Thư viện cơ sở B của trường”	9
Hình 6: Biểu đồ cột thể hiện số lượng chữ ở mỗi câu bình luận.....	11
Hình 7: Biểu đồ cột thể hiện số lượng câu bình luận theo điểm đánh giá	11
Hình 8: Biểu đồ tròn thể hiện tỷ lệ các câu bình luận theo điểm đánh giá.....	12
Hình 9: Bảng dữ liệu sau khi thực hiện tiền xử lý	13
Hình 10: Biểu đồ cột thể hiện 10 token phổ biến nhất của bộ dữ liệu theo từng ngôn ngữ.....	13
Hình 11: Kết quả 5 câu có độ dài lớn nhất trong tiếng Anh	14
Hình 12: Kết quả 5 câu có độ dài lớn nhất trong tiếng Việt	14
Hình 13: Đại diện một số kết quả POS Tagging cho các nhận xét tiếng Việt	15
Hình 14: Đại diện một số kết quả POS Tagging cho các nhận xét tiếng Anh	16
Hình 15: Biểu đồ cột thể hiện tần suất xuất hiện loại từ của các bình luận	16
Hình 16: Biểu đồ WordCloud các danh từ xuất hiện nhiều nhất	17
Hình 17: Các bình luận tiêu cực về chủ đề Giảng viên.....	18
Hình 18: Tuple với các cặp token đi với giá trị True và nhãn “sentiment”	23
Hình 19: Thực hiện dự đoán ngôn ngữ tiếng Anh Naive Bayes	24
Hình 20: Thực hiện dự đoán ngôn ngữ tiếng Việt Naive Bayes	24
Hình 21: Tuple với các cặp token đi với giá trị True và nhãn “sentiment”	25
Hình 22: Vòng lặp đạt kết quả tối ưu.....	26
Hình 23: Bảng kết quả đánh giá.....	26
Hình 24: Ma trận nhầm lẫn cho tập test.....	27
Hình 25: Kết quả thực nghiệm dự đoán cho 5 câu đánh giá bất kỳ	27
Hình 26: Biểu đồ Histogram thể hiện độ dài của các câu	28
Hình 27: Kiến trúc model xây dựng (từ bước input tới bước kết hợp mô hình)	29
Hình 28: Kiến trúc model xây dựng (từ bước kết hợp mô hình tới bước output)	31
Hình 29: Kết quả classification report từ mô hình.....	32
Hình 30: Kết quả classification report từ mô hình gán trọng số nhãn	32
Hình 31: Biểu đồ cột cụm thể hiện các chỉ số đánh giá qua từng mô hình.....	33
Hình 32: Biểu đồ cột thể hiện thời gian huấn luyện mô hình của từng mô hình	34
Hình 33: Ứng dụng Flask để xây dựng trên mô hình học sâu.....	35
Hình 34: Xây dựng giao diện web bằng html:5	36
Hình 35: Giới thiệu cách khởi chạy demo	36
Hình 36: Khởi chạy development server xây dựng bằng Flask	37
Hình 37: Website demo lúc vừa khởi chạy development server.....	37
Hình 38: Website demo sau khi thực nghiệm với bình luận bất kỳ	38

PHỤ LỤC

Mã nguồn

https://github.com/lemonade140403/NLP_Sentiment_Analysis

Bảng phân công

SINH VIÊN	NHIỆM VỤ	ĐÁNH GIÁ
Lê Thị Ngọc Ánh	<ul style="list-style-type: none">Tổng quan đề tài (C1)Tiền xử lý dữ liệu và EDAXây dựng lý thuyết và thuật toán mô hình Naïve BayesKết luận và đánh giá (C6)	100%
Trần Phạm Hải Nam	<ul style="list-style-type: none">Tổng quan bộ dữ liệu (C3)Cào dữ liệu từ Google MapsPhân tích dữ liệu bằng POS taggingXây dựng lý thuyết và thuật toán mô hình CNN kết hợp BiLSTMĐánh giá so sánh trực quanXây dựng demoTổng hợp báo cáo	100%
Lý Minh Nguyên	<ul style="list-style-type: none">Tiền xử lý dữ liệuXây dựng lý thuyết và thuật toán mô hình MaxentCác hướng cải tiến (C6)Tổng hợp và chỉnh sửa mã nguồn	100%

DANH MỤC THAM KHẢO

- [1] Tsaniya, Hilya & Rosadi, Revlita & Abdullah, A. (2021). *Sentiment analysis towards Jokowi`s government using twitter data with convolutional neural network method*. Retrieved from researchgate.net/publication/348346020
- [2] Zhang, Ye & Wallace, Byron. (2015). *A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification*. Retrieved from researchgate.net/publication/282906526
- [3] Tsung-Yi Lin & Priya Goyal & Ross Girshick & Kaiming He & Piotr Dollar. (2017). *Computer Vision Foundation. Focal Loss for Dense Object Detection*. Retrieved from openaccess.thecvf.com/content_ICCV_2017
- [4] Sultan, Daniyar & Toktarova, Aigerim & Zhumadillayeva, Ainur & Aldeshov, Sapargali & Mussiraliyeva, Shynar & Beissenova, Gulbakhram & Tursynbayev, Abay & Baenova, Gulmira & Imanbayeva, Aigul. (2023). *Cyberbullying-related Hate Speech Detection Using Shallow-to-deep Learning*. Retrieved from researchgate.net/publication/366762592
- [5] Zhang, Lei & Wang, Shuai & Liu, Bing. (2018). *Deep Learning for Sentiment Analysis : A Survey*. Retrieved from researchgate.net/publication/322694910
- [6] Shashank Kalluri. (2023 Jan). *Deep Learning Based Sentiment Analysis*. Retrieved from diva-portal.org/diva2:1741487
- [7] Noor Saeed. (2023). *Sentiment-Analysis-Mahcine-Learning-NLP-Project*. GitHub repository. Retrieved from github.com/Sentiment-Analysis-Mahcine-Learning-NLP-Project
- [8] ChauLu38. (2021). *Shopee_Comments_Sentiment*. GitHub repository. Retrieved from github.com/Shopee_Comments_Sentiment
- [9] JNoether. (2021). *Multi_Intent_Recognition*. GitHub repository. Retrieved from github.com/Multi_Intent_Recognition
- [10] Alice Zhao. (2022). *nlp-in-python-tutorial*. GitHub repository. Retrieved from github.com/nlp-in-python-tutorial
- [11] HoLuan. (2023). *RNN-with-Numpy*. GitHub repository. Retrieved from github.com/RNN-with-Numpy
- [12] qbingking. (2019). *senti-ana*. GitHub repository. Retrieved from github.com/senti-ana
- [13] Thu Hương Orianna. (2023). *NLP_Python-Sentiment-Analysis-Project-*. Retrived from github.com/NLP_Python-Sentiment-Analysis-Project-
- [14] Lavanya Gupta. (2021, Jan 28). *Focal Loss — What, Why, and How?*. Retrieved from medium.com/focal-loss-what-why-and-how-df6735f26616

- [15] Monkey Learn. (n.d.). *Sentiment Analysis: A Definitive Guide*. Retrieved from monkeylearn.com/sentiment-analysis
- [16] Denny. (2015, Nov 7). *Understanding Convolutional Neural Networks for NLP*. Retrieved from dennybritz.com/understanding-convolutional-neural-networks-for-nlp
- [17] Vasilis Vryniotis. (2013, Oct 20). *Machine Learning Tutorial: The Max Entropy Text Classifier*. Retrieved from blog.datumbox.com/machine-learning-tutorial-the-max-entropy-text-classifier
- [18] Enes Zvornicanin. (2023, June 8). *Differences Between Bidirectional and Unidirectional LSTM*. Retrieved from baeldung.com/bidirectional-vs-unidirectional-lstm
- [19]. Christopher Olah. (2015, Aug 27). *Understanding LSTM Networks*. Retrieved from colah.github.io/2015-08-Understanding-LSTMs
- [20] Do Dang Hung. (2021, Sep 14). *Tiến xử lý dữ liệu văn bản với NLTK*. Retrived from viblo.asia/tien-xu-li-du-lieu-van-ban-voi-nltk
- [21] VBD. (2022, Aug 17). *Xử lý ngôn ngữ tự nhiên: Bài toán & công cụ bạn nên biết*. Retrived from vinbigdata.com/xu-ly-ngon-ngu-tu-nhien-bai-toan-cong-cu-ban-nen-biet
- [22] Nga Vu. (2022, Apr 8). *Tìm hiểu Naive Bayes Classification - Phần 1*. Retrieved from 200lab.io/tim-hieu-naive-bayes-classification-phan-1