Phân tích cảm xúc qua một câu bình luận đồ ăn

Nguyễn Phúc Đạt - 18520573 - CS114.K21.KHTN

Link Github:

https://github.com/DatDoc/CS114.K21.KHTN

I. Đặt vấn đề

Với sự phát triển của các trang buôn bán điện tử ngày nay thì việc thu thập đánh giá từ các bình luận của người dùng là rất cần thiết. Các bình luận của người dùng giúp cho các trang web dễ dàng lọc, đề xuất các mặt hàng, địa điểm phù hợp với từng người dùng và đánh giá chất lượng dịch vụ của các đối tác. Với hàng chục nghìn cho tới hàng trăm nghìn bình luận mỗi ngày hiện nay thì việc phân loại (tốt, xấu, ngon, dở) cho các bình luận từ người dùng không phải là điều dễ dàng cần đòi hỏi rất nhiều nhân lực. Cùng sự phát triển của AI, hiện nay chúng ta có thể giải quyết bài toán này bằng các thuật toán và mô hình học sâu với độ chính xác cao tương đương với con người.

II. Bài toán cụ thể

Trong assignment này, bài toán được tiếp cận là phân tích cảm xúc (Sentiment Analysis) tiêu cực/ tích cực thông qua tập dữ liệu foody chứa các bình luận được crawl trực tiếp từ https://www.foody.vn/.

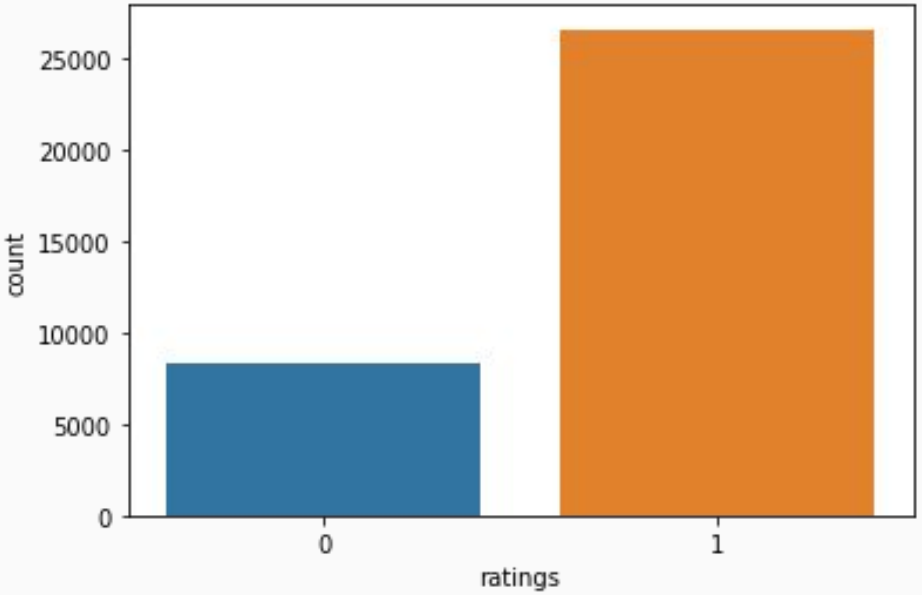
- Input: 1 câu bình luận về thức ăn
- Output: tiêu cực / tích cực

III. Dữ liệu

- Dataset của bài toán này chứa 61.870 bình luận và điểm đánh giá của mỗi bình luận đó trên thang điểm từ 0.0 đến 10.0, dữ liệu được crawl từ hai thành phố lớn là Hồ Chí Minh và Hà Nội. Sử dụng công cụ Web Scraper để crawl dữ liệu.
- Tuy nhiên, để bộ dữ liệu phù hợp với bài toán là đánh giá cảm xúc tiêu cực hay tích cực, nên trong bước tiền xử lý dữ liệu, các bình luận được đánh giá >= 8.0 thì sẽ được gán nhãn tích cực (1), các bình luận được đánh giá <= 5.0 thì được gán nhãn tiêu cực (0). Do đó số lượng bình luận trong bộ dữ liệu sẽ bị giảm xuống còn 34.985 bình luận bao gồm 8366 bình luận tiêu cực và 26619 bình luận tích cực.

III. Dữ liệu

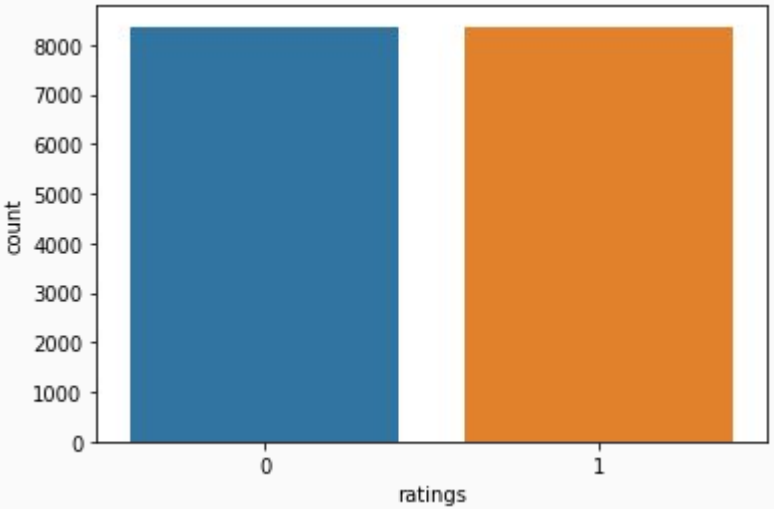
Dữ liệu sau khi đã xử lý nhóm bình luận > 5.0 và <8.0



Positive: 26619
Negative: 8366

III. Dữ liệu

Nhìn vào biểu đồ trên ta nhận thấy việc dữ liệu bị imbalance **Undersampling** là kỹ thuật được áp dụng trong bài toán này để khử imbalance.



Positive: 8366
Negative: 8366

IV. Tiền xử lý dữ liệu

Các bước tiền xử lý dữ liệu

1. Loại bỏ các ký tự lặp lại
2. Chuẩn hóa các ký tự thành chữ in thường
3. Chuẩn hóa dấu câu và các từ cảm xúc (sentiment word) trong Tiếng Việt.
4. Chuẩn hóa các emojis về hai loại: tích cực và tiêu cực.
5. Loại bỏ các ký tự trong HTML.
6. Loại bỏ các punctuations và các emoji vô nghĩa.
7. Tách từ - sử dụng thư viện underthesea.
8. Tăng cường dữ liệu (data augmentation): bằng cách thêm các bình luận không dấu được generate từ dữ liệu gốc.

IV. Tiền xử lý dữ liệu

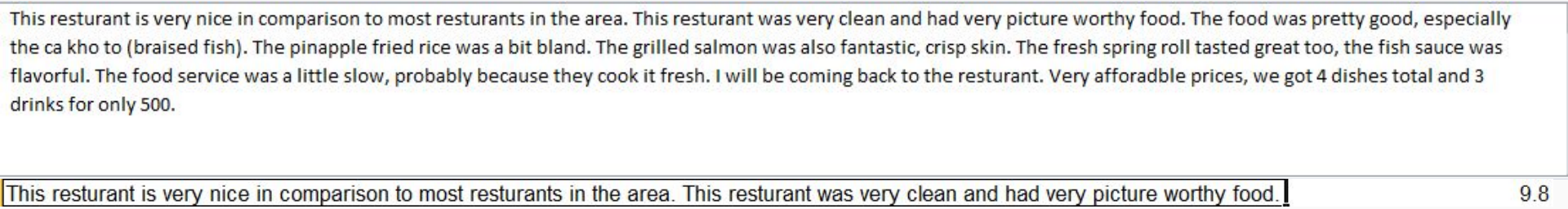
Các khó khăn:

- Chưa xử lý được các câu văn tiếng nước ngoài (vd: I love this dish, 나는이 음식을 좋아한다, 私はこの食べ物が大好きです)
- Chưa xử lý được các bình luận bị dán nhãn sai
  Ví dụ: “ Chỗ này đẹp quá!!” nhưng lại gán cho điểm số chỉ có 5.0.

IV. Tiền xử lý dữ liệu

Các khó khăn:

- Chưa xử lý được các câu văn tiếng nước ngoài (vd: I love this dish, 나는이 음식을 좋아한다, 私はこの食べ物が大好きです)



- Chưa xử lý được các bình luận bị dán nhãn sai
  Ví dụ: “ Chỗ này đẹp quá!!” nhưng lại gán cho điểm số chỉ có 5.0.

V. Rút trích đặc trưng và chọn model

- TF - IDF là kỹ thuật được áp dụng cho các thuật toán trong thư viện sklearn như Multinomial NB, Logistic Regression, Linear SVC.
- Word embedding: ở đây sử dụng thư viện keras của tensorflow để custom một deep learning model với số chiều vector của một từ là 100, số lượng từ thống nhất trong một câu là 250 từ.

```
model_dl = tf.keras.Sequential()
model_dl.add(tf.keras.layers.Embedding(vocab_size, embedding_dim, input_length = max_length))
model_dl.add(tf.keras.layers.Flatten())
model_dl.add(tf.keras.layers.Dense(10, activation="relu"))
model_dl.add(tf.keras.layers.Dense(1, activation="sigmoid"))
model_dl.compile(loss = 'binary_crossentropy', optimizer='adam', metrics=['accuracy'])
model_dl.summary()
```

```
vocab_size = 10000
embedding_dim = 100
max_length = 250
```

VI. Training

Multinomial NB

	precision	recall	f1-score	support
0	0.914	0.854	0.883	5106
1	0.859	0.916	0.887	4934
micro avg	0.885	0.885	0.885	10040
macro avg	0.886	0.885	0.885	10040
weighted avg	0.887	0.885	0.885	10040

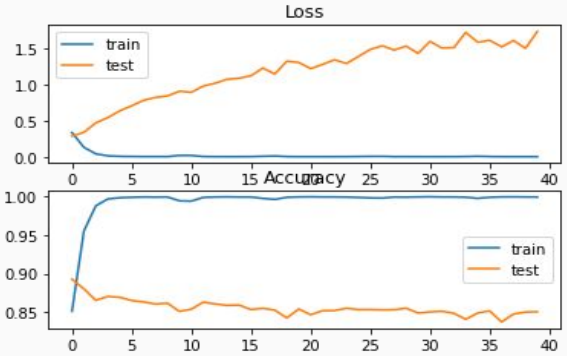
Logistic Regression

	precision	recall	f1-score	support
0	0.903	0.881	0.892	5106
1	0.880	0.902	0.891	4934
micro avg	0.891	0.891	0.891	10040
macro avg	0.891	0.891	0.891	10040
weighted avg	0.891	0.891	0.891	10040

Linear SVC

	precision	recall	f1-score	support
0	0.881	0.869	0.875	5106
1	0.866	0.878	0.872	4934
micro avg	0.873	0.873	0.873	10040
macro avg	0.873	0.873	0.873	10040
weighted avg	0.873	0.873	0.873	10040

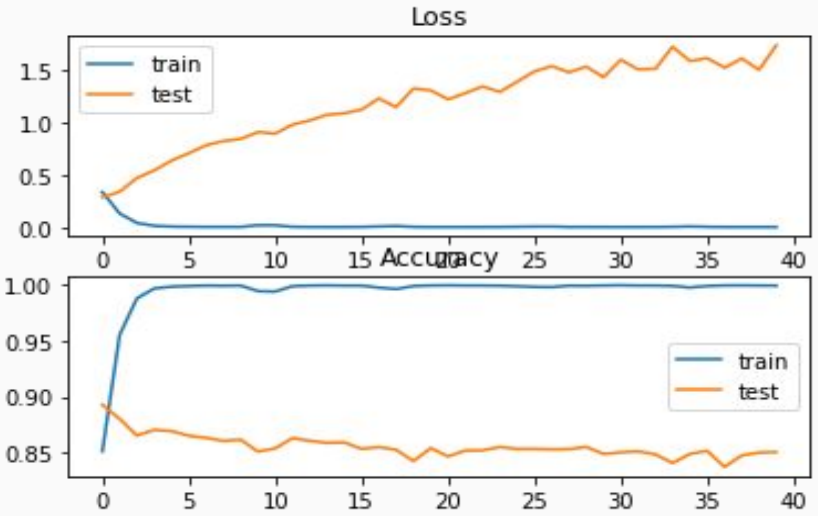
Deep learning model



Accuracy: 0.9992
Validation accuracy: 0.8507

VII. Đánh giá

- MultinomialNB cho kết quả tốt nhất trong bốn model với bộ dữ liệu balance: 0.914.
- Deep learning model cho kết quả thấp nhất với 0.8507



Đánh giá quá trình huấn luyện deep learning model:

- Ở đồ thị loss, khoảng cách 2 đường train và test ngày càng xa nhau khi đi về phía cuối biểu đồ. Có thể xảy ra overfit nếu tăng thêm số lượng epochs
- Ở đồ thị accuracy, khoảng cách 2 đường train và test khá ổn định

VIII. Tuning các tham số

Sau khi tuning thì Logistic Regression kết quả tốt nhất

	precision	recall	f1-score	support
0	0.903	0.881	0.892	5106
1	0.880	0.902	0.891	4934
micro avg	0.891	0.891	0.891	10040
macro avg	0.891	0.891	0.891	10040
weighted avg	0.891	0.891	0.891	10040

Trước khi tuning

	precision	recall	f1-score	support
0	0.920	0.888	0.904	5106
1	0.888	0.921	0.904	4934
micro avg	0.904	0.904	0.904	10040
macro avg	0.904	0.904	0.904	10040
weighted avg	0.905	0.904	0.904	10040

Sau khi tuning

IX. Dự đoán trên dữ liệu mới

Tập dữ liệu test cũng được crawl từ foody nhưng địa điểm là ở Quảng Nam.

	precision	recall	f1-score	support
0	0.755	0.846	0.798	1000
1	0.958	0.928	0.943	3798
micro avg	0.911	0.911	0.911	4798
macro avg	0.856	0.887	0.870	4798
weighted avg	0.916	0.911	0.912	4798

Multinomial NB

	precision	recall	f1-score	support
0	0.742	0.888	0.809	1000
1	0.969	0.919	0.943	3798
micro avg	0.912	0.912	0.912	4798
macro avg	0.856	0.903	0.876	4798
weighted avg	0.922	0.912	0.915	4798

Logistic Regression

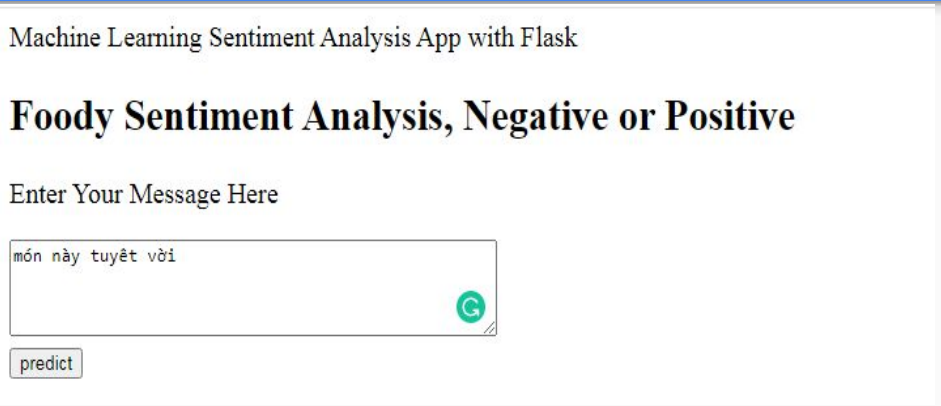
	precision	recall	f1-score	support
0	0.721	0.883	0.794	1000
1	0.967	0.910	0.938	3798
micro avg	0.905	0.905	0.905	4798
macro avg	0.844	0.897	0.866	4798
weighted avg	0.916	0.905	0.908	4798

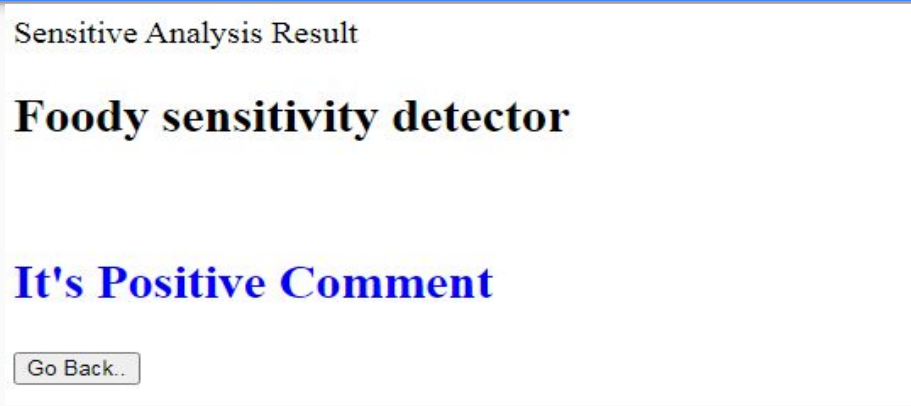
Linear SVC

	precision	recall	f1-score	support
0	0.654	0.819	0.727	1000
1	0.949	0.886	0.916	3798
micro avg	0.872	0.872	0.872	4798
macro avg	0.801	0.852	0.822	4798
weighted avg	0.887	0.872	0.877	4798

Deep learning model

X. Deploy webapp





Sử dụng thư viện flask-ngrok để deploy model lên webapp
Video đầy đủ: https://drive.google.com/file/d/1pmDQs7bGtT7VqV5OL3OSJ3Z2bpj85nH_/view?usp=sharing