ĐHQG TP. Hồ Chí Minh
Trường ĐH Công Nghệ Thông Tin
-oOo-

ĐỒ ÁN CUỐI KỲ

PHÂN TÍCH CẢM XÚC TỪ BÌNH LUẬN SẢN PHẨM SHOPEE

Môn học:	Tư duy tính toán – **CS117.L22.KHCL**
Giảng viên lý thuyết:	Ngô Đức Thành
Giảng viên thực hành:	
Sinh viên thực hiện:	Lương Phạm Bảo 19521242
	Nguyễn Phú Lộc 19520687
	Đoàn Duy Ân 19521172

TP. Hồ Chí Minh, 29 tháng 08 năm 2021

I. Phân tích cảm xúc cho bình luận sản phẩm

1. Lý do chọn bài toán

Bán hàng online là xu thế công nghệ của ngày nay, gần như mọi gia đình đều sẽ mua ít nhất một món hàng online mỗi tuần. Tuy nhiên khó có thể mà kiểm định được chất lượng có đảm bảo hay không. Đặc biệt với các mặt hàng đắt tiền được bày bán nhan nhản ở khắp mọi nơi trên internet, và việc lựa chọn mua ở đâu, mua hãng gì cho tốt trở thành mối quan tâm lớn cho người dùng.

Một trong những cách để quyết định có nên mua hay không là dựa vào đánh giá từ những người đã mua trước, tuy nhiên số lượng đánh giá rất lớn, không có nhân lực để thống kê được hết. Vì thế áp dụng machine Learning nói riêng cũng như cách giải quyết vấn đề dựa trên máy tính nói chung trong việc phân loại đánh giá của khách hàng là một việc đơn giản và hiệu quả và tiết kiệm chi phí.

Ứng dụng bài toán: Thông qua việc tự động phân loại và dự đoán chất lượng các câu bình luận có thể tìm ra các sản phẩm được review, đáng tiền để mua, giúp người mua có thể đánh giá phân biệt sản phẩm một cách dễ dàng hơn

2. Xác định bài toán

Input là gì?

- Có 1 input duy nhất.
- Một câu comment có định dạng text về một bình luận về một sản phẩm(dữ liệu ở dưới dạng Tiếng Việt có dấu),1 câu comment có thể chứa nhiều câu .
- Chú ý:câu bình luận có độ dài không quá 200 từ (thông thường các câu có độ dài 200 từ thường là spam).

Output là gì?

- Có 1 output duy nhất.
- Output có đinh dạng text.
- Giá trị của output gồm một trong hai loại (Positive,Negative) .
- Positive: Câu bình luận mang tính tích cực,đánh giá cao về sản phẩm VD: Shop giao hàng nhanh, đóng gói cẩn thận.
- Negative: Câu bình luận mang tính tiêu cực,chê bai,không hài lòng về sản phẩm
  VD: Khô gà ăn ỉu, cay nhiều chứ ko phải cay vừa. Ko đc ngon như lần chị m mua.

1. Evaluation

Cách thu thập dữ liệu: Data được thu thập thông qua 2 cách: crawl các comment về sản phẩm từ trang sản phẩm của shopee và lấy các tập dữ liệu được thu thập từ các cuộc thi của shopee,dữ liệu được crawl sẽ được 2 người đánh nhãn độc lập(nhãn có tỉ lệ đồng thuận trên 70%).

Tiêu chí đánh giá: Được đánh giá bằng kết quả dự đoán đúng trên những câu bình luận khác(các câu chưa có trong bộ dữ liệu, các câu comment mới cho một sản phẩm) ,=s ố lượng câu dự đoán đúng/(trên)số câu muốn dự đoán. Ngoài ra có thể sử dụng một đơn vị đo trong Máy học như F1 score,accuracy Dữ liệu thử nghiệm được tổng hợp lại và gán nhãn sẵn (tỉ lệ đồng thuận trên 85%) nhưng không đưa vào tập huấn luyện (cả 3 bạn đều đánh nhãn độc lập và có thống nhất lại để có tỉ lệ đồng nhất cao).

2. Mô tả dữ liệu

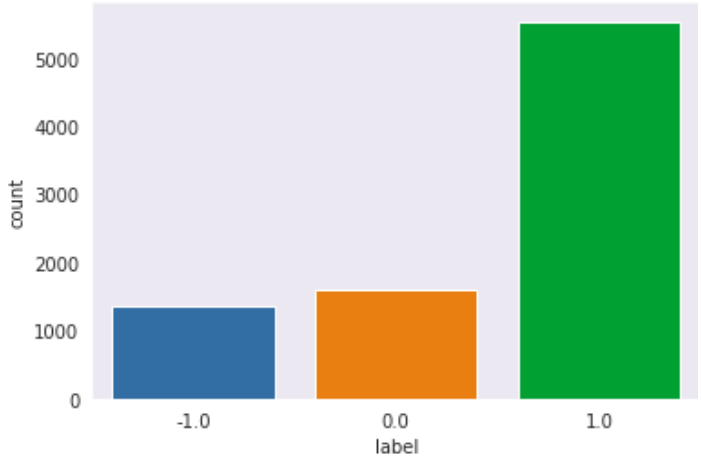
Nhóm chúng em sẽ sử dụng dữ liệu từ 2 nguồn:

Craw data: Nhóm em sẽ sử web crapping từ thư viện Beatiful Soup của python để craw các comment của shopee ở một số mặt hàng.

Data thu thập: Lựa chọn một dataset từ cuộc thi của kaggle của shopee và một số bộ dataset public khác.

Dữ liệu comment craw được khoảng 20000 comment. Do khó khăn về mặt thời gian nên chúng em sẽ lựa chọn 6000 câu tốt từ 20000 câu trên để label, sau đó lọc ra một số câu để được khoảng 5000 câu và gộp với dữ liệu với dữ liệu chúng em kiếm được để có được 8497 câu bình luận.

Phân phối dữ liệu:



1 : 5538
0 : 1598
-1: 1361

Phân chia dữ liệu

Nhóm em sẽ chia dữ liệu thành 2 tập train và test theo tỉ lệ 8: 2 để huấn luyện mô hình(tập train lại được chia theo tỉ lệ 3:1 ứng với tập train và validation).

Ta thấy rằng dữ liệu bị mất cân bằng nên nhóm em sẽ cân nhắc sử dụng thêm F1 score để đánh giá bài toán một cách chính xác hơn.

Cấu trúc giải quyết cho bai toán

Để giải quyết bài toán nhóm em đã phân chia bài toán thà nh cấu trúc gồm 3 phần chính:

- Data
- Training
- Demo

Với 3 phần trên nhóm em thấy rằng có thể áp dụng các pattern của việc giải quyết một bài toán máy học (cụ thể là bài toán phân loại)

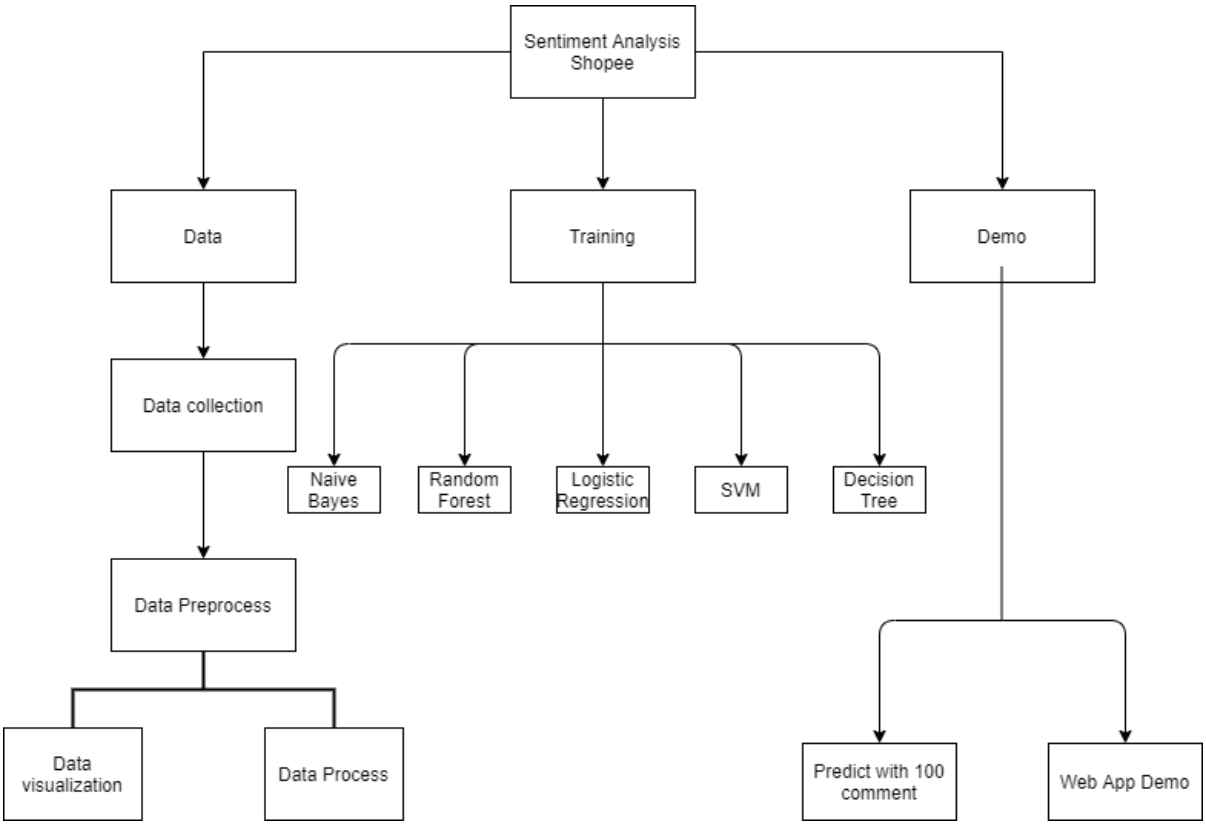
Sau khi phân chia bài toán thành 3 vấn đề nhỏ hơn, chúng em sẽ thực hiện giải quyết vấn đề bằng các bước tương tự khi giải quyết một vấn đề máy học(các công đoạn để giải quyết một bài toán máy học điển hình)

Ngoài ra ở mỗi bước thực hiện, sẽ có một số bước nhỏ kèm theo(rõ hơn ở phần Graphic Organizer)

Các vấn đề ở node trên tuy chưa là một vấn đề quen đơn giản nhưng ta có thể sử dụng các pattern recognition về bài toán text classification để giải quyết (các phần ở các node lá là các công đoạn được xem là cơ bản trong các bài toán có sử dụng máy học

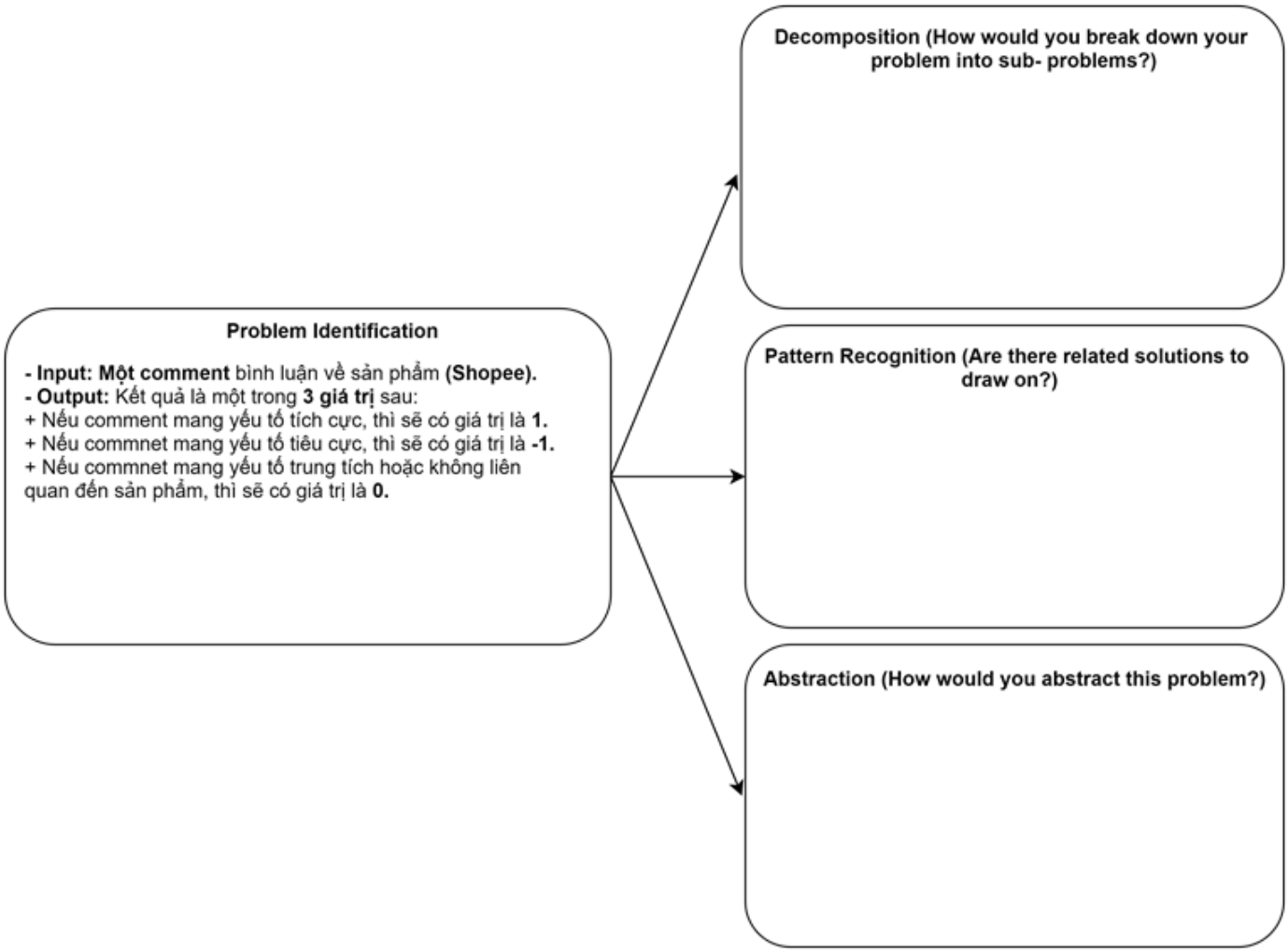
Sơ đồ giải quyết :

Data → Training → Demo

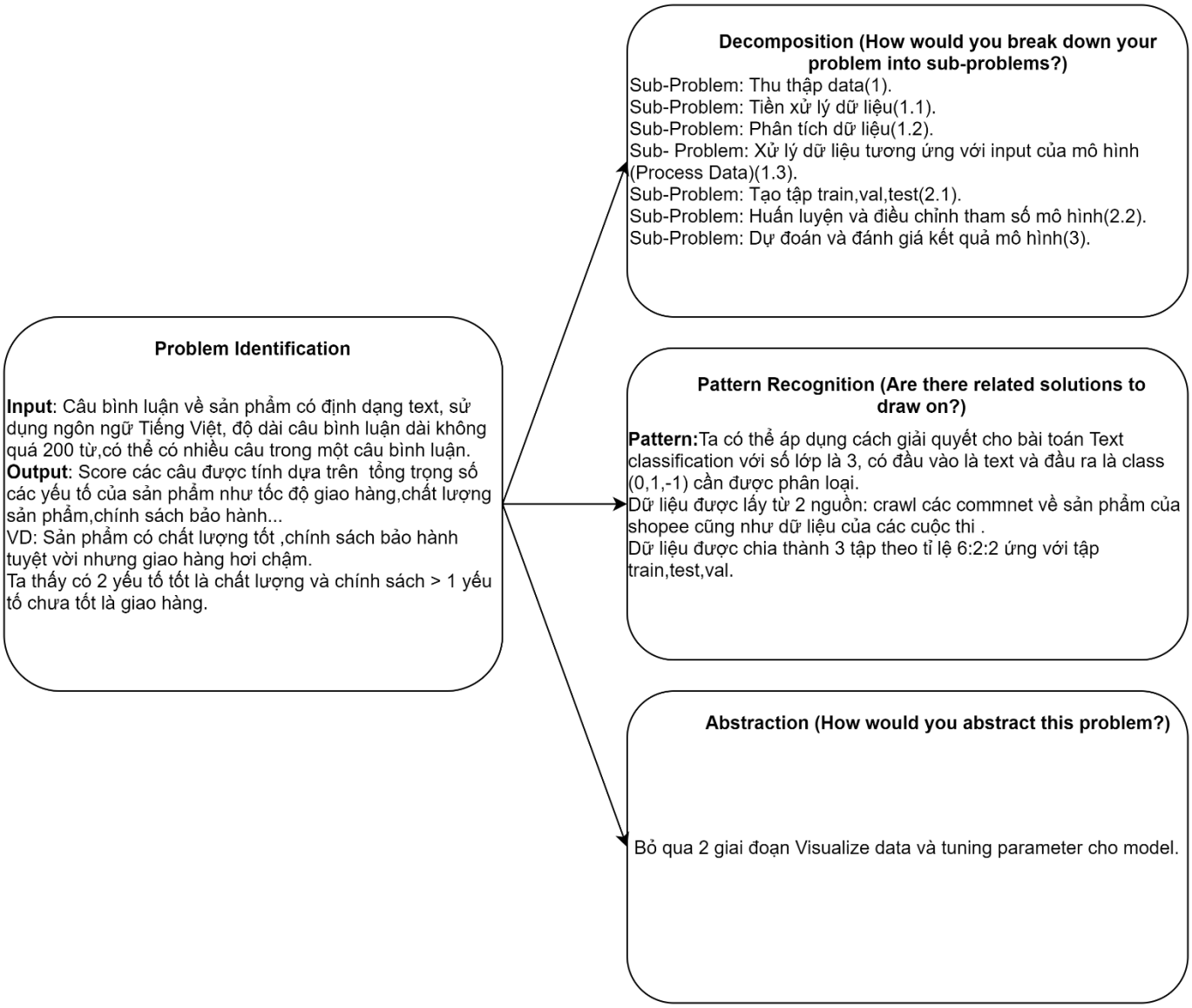


I. Graphic Organizer

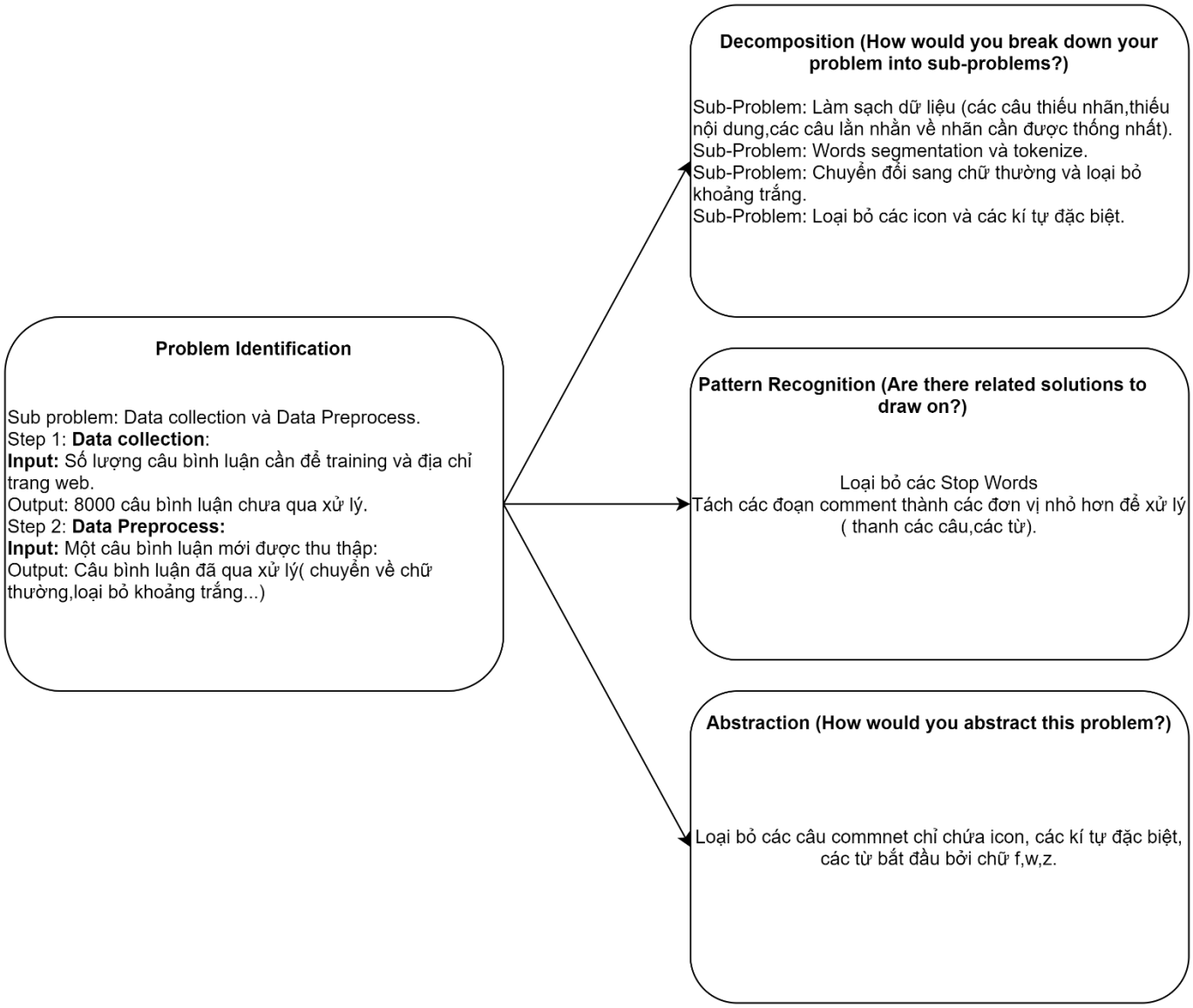
1. Iteration 1



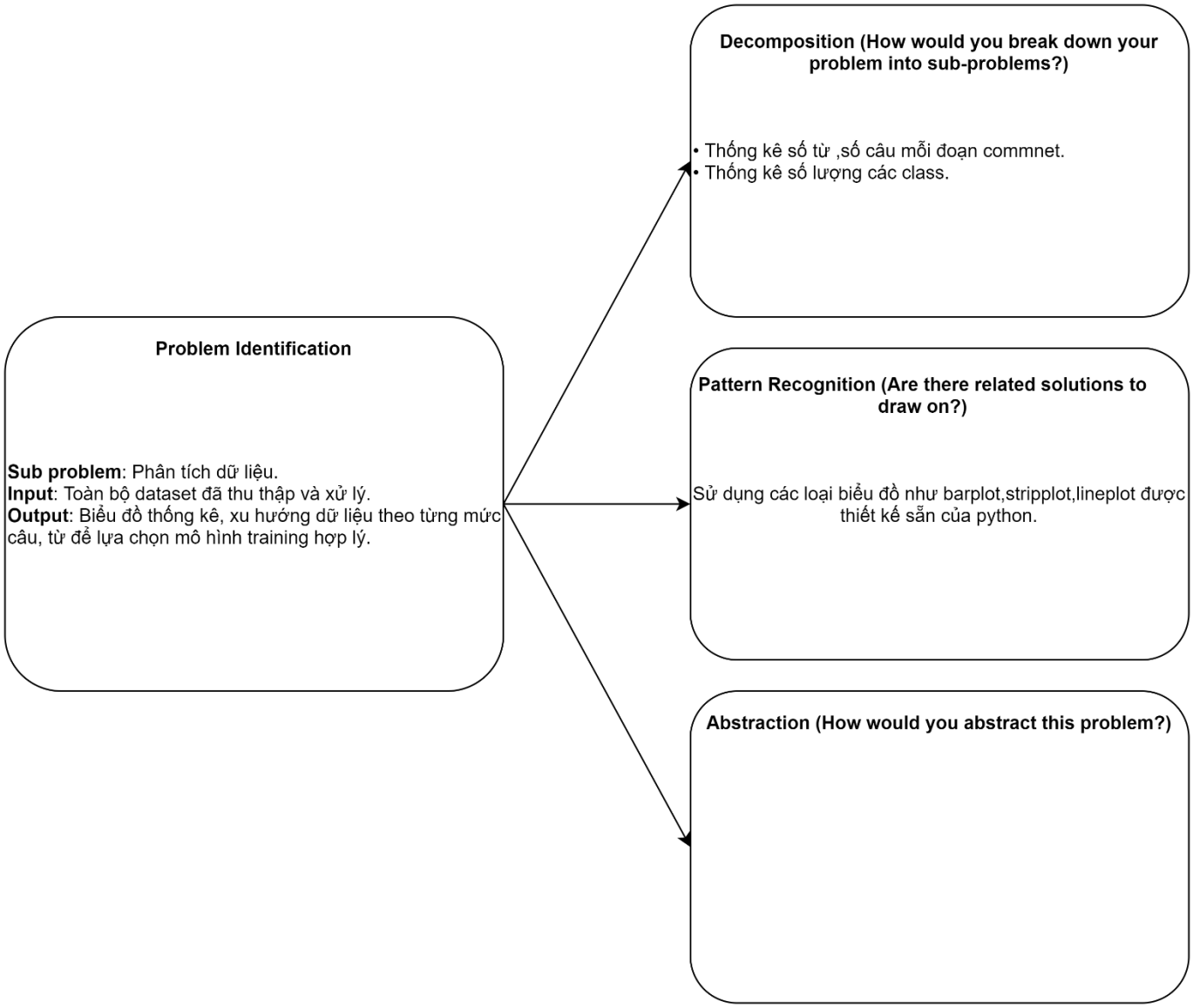
2. Iteration 2



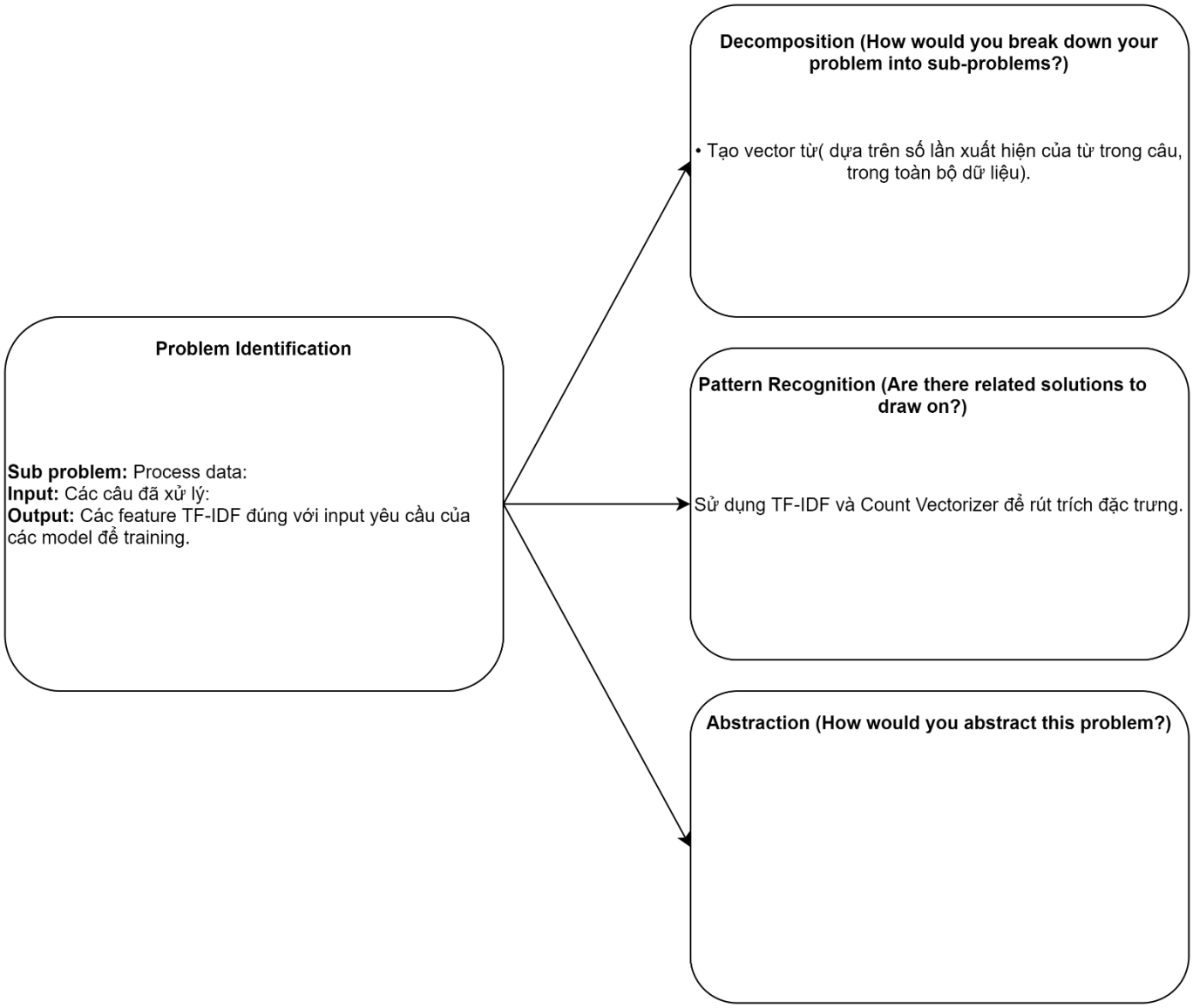
3. Iteration 3



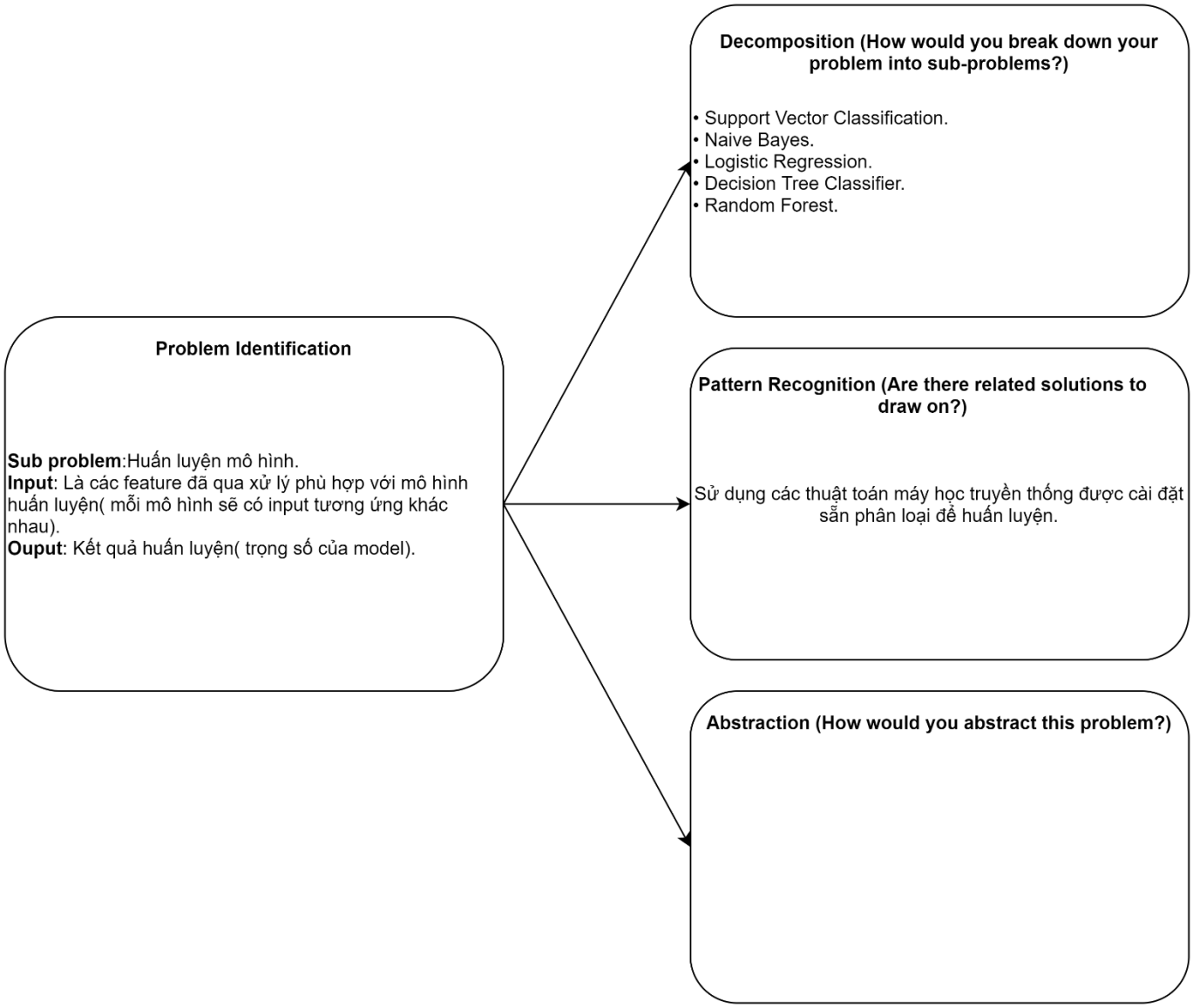
4. Iteration 4



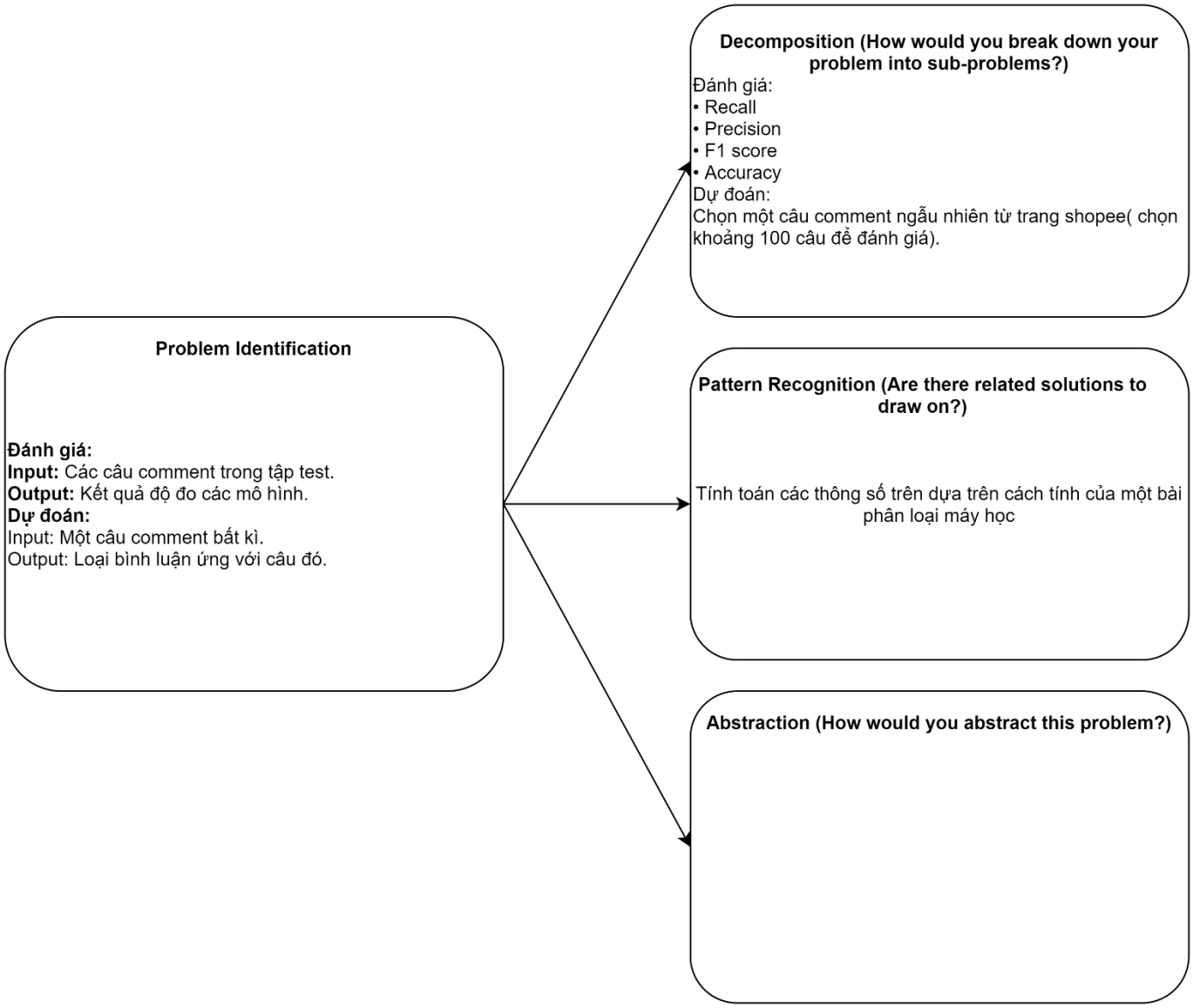
5. Iteration 5



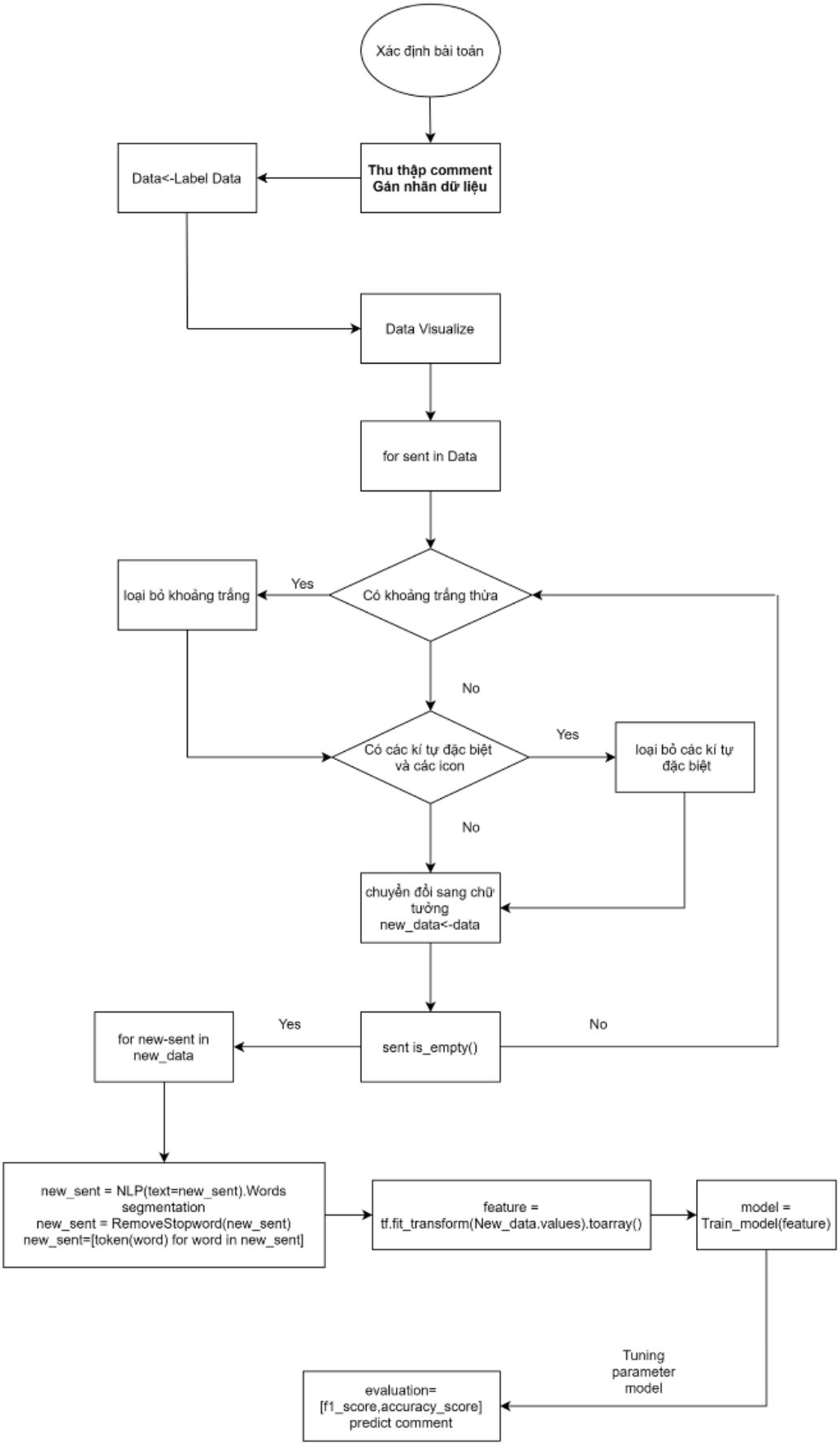
6. Iteration 6



7. Iteration 7



II. FlowChart



III. Mô tả các kĩ thuật cho bài toán

1. TF-IDF

ĐN: TF-IDF là trọng số của một từ trong văn bản thu được qua thống kê ,thể hiện mức độ quan trọng của từ này trong một văn bản, mà bản thân văn bản đang xét cũng nằm trong một tập hợp các văn bản.

Tf- term frequency: Dùng để ước lượng tần xuất xuất hiện của từ trong văn bản. Tuy nhiên với mỗi văn bản thì có độ dài khác nhau, vì thế số lần xuất hiện của từ có thể nhiều hơn . Vì vậy số lần xuất hiện của từ sẽ được chia độ dài của văn bản (tổng số từ trong văn bản đó).

TF(t, d) = (số lần từ t xuất hiện trong văn bản d) / (tổng số từ trong văn bản d)

IDF- Inverse Document Frequency: Dùng để ước lượng mức độ quan trọng của từ đó như thế nào. Khi tính tần số xuất hiện tf thì các từ đều được coi là quan trọng như nhau. Tuy nhiên có một số từ thường được được sử dụng nhiều nhưng không quan trọng để thể hiện ý nghĩa của đoạn văn , ví dụ:
Từ nối: và, nhưng, tuy nhiên, vì thế, vì vậy, …
Giới từ: ở, trong, trên, …
Từ chỉ định: ấy, đó, nhỉ, …
Vì vậy ta cần giảm đi mức độ quan trọng của những từ đó bằng cách sử dụng IDF :

IDF(t, D) = log_e(Tổng số văn bản trong tập mẫu D/ Số văn bản có chứa từ t)

Tf-idf có trọng số cao đối với các từ xuất hiện thường xuyên trong tài liệu hiện tại, nhưng hiếm khi xuất hiện trong tập tài liệu tổng thể (document collection), cho thấy rằng từ đó đặc biệt liên quan đến tài liệu này. Đối với mỗi từ xuất hiện trong câu, ta lấy tổng số lượng của từ đó trong tài liệu nhân với nghịch đảo tần suất của tài liệu đó trên tập tài liệu tổng thể.

2. Word Segmentation

Word Segmentation là quá trình phân chia văn bản đã viết thành các đơn vị có nghĩa, chẳng hạn như từ, câu hoặc chủ đề. Thuật ngữ này áp dụng cho cả các quá trình tinh thần được sử dụng bởi con ngườikhi đọc văn bản và các quy trình nhân tạo được thực hiện trong các lĩnh xực xử lý ảnh, xử lý ngôn ngữ tự nhiên.

3. Stop word

Với các bài toán NLP nói chung, ta cần phải rút trích các thông tin thông qua các đặc trưng từ các câu, các từ trong văn bản. Thông thường ta sẽ lấy các từ có số lần xuất hiện nhiều trong văn bản để đánh giá, tuy nhiên có nhưng câu dù có số lần xuất hiện nhiều nhưng lại không mang lại thông tin nào cụ thể. Vì vậy trước khi tính tần số của các từ trong văn bản, ta cần loại bỏ bớt các từ như thế ra khỏi văn bản nhằm giúp cho kết quả huấn luyện và dự đoán của mô hình trở nên tốt hơn.

VD: **sản phẩm có mẫu mã đẹp, giao hàng nhanh nhưng sản phẩm có bị trầy → sản phẩm mẫu mã đẹp, giao hàng nhanh sản phẩm trầy.**

4. Model sử dụng

a) **MultinomialNB:** Phân loại văn bản sử dụng Naïve Bayes sẽ được dựa trên một "bag of word", trong đó, mỗi từ vựng sẽ được đánh số là 0 – với những từ không có trong văn bản đang xem xét và 1 – với những từ xuất hiện trong văn bản đang xem xét.

b) **LogictisRegression:** Phương pháp hồi quy logistic là một mô hình hồi quy nhằm dự đoán giá trị đầu ra rời rạc (discrete target variable) y ứng với một véc-tơ đầu vào x. Việc này tương đương với chuyện phân loại các đầu vào x vào các nhóm y tương ứng.

c) **SVM:** SVM là một thuật toán máy học giám sát, nó có thể sử dụng cho cả việc phân loại hoặc đệ quy. Tuy nhiên nó được sử dụng chủ yếu cho việc phân loại. Trong thuật toán này, chúng ta vẽ đồi thị dữ liệu là các điểm trong n chiều (ở đây n là số lượng các tính năng bạn có) với giá trị của mỗi tính năng sẽ là một phần liên kết. Sau đó chúng ta thực hiện tìm "margin" phân chia các class. Margin, nó chỉ hiểu đơn giản là 1 đường thẳng có thể phân chia các lớp ra thành hai phần riêng biệt .

d) **Decision tree:** Là một mô hình ra quyết định dựa trên các câu hỏi. Mô hình này có tên là cây quyết định (decision tree), sử dụng các câu bình luận để học cách quyết định giữa các lựa chọn về nhãn.

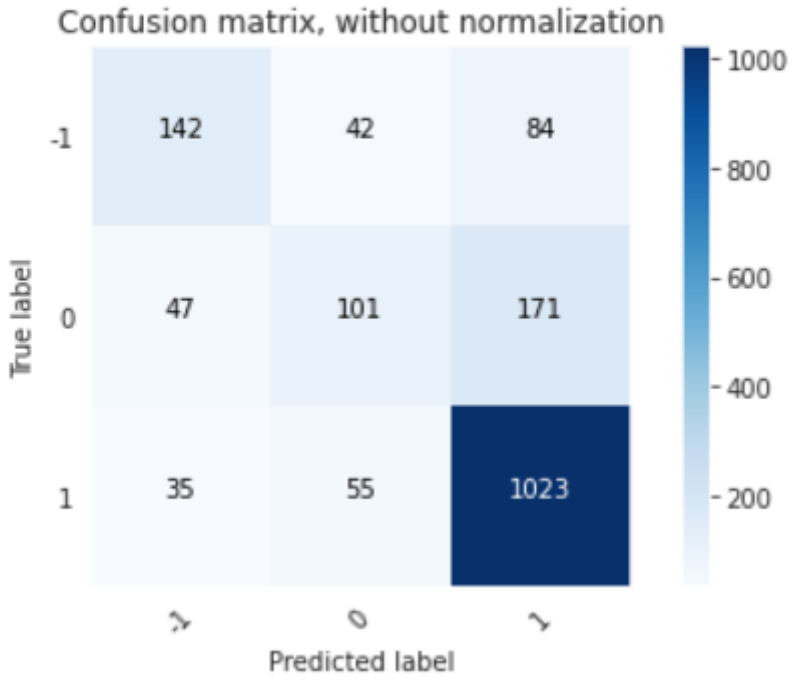
e) **Random Forest:** Đây là phương pháp xây dựng một tập hợp rất nhiều cây quyết định và sử dụng phương pháp voting để đưa ra quyết định về biến target cần được dự báo.

IV. Kết quả mô hình

1. Kết quả mô hình Random Forest

```
Model RandomForestClassifier
Train score:  0.9882301015153744
Test score:  0.7447058823529412
F1 score:  0.6078867826918867
```

```
Confusion matrix, without normalization
[[ 142   42   84]
 [  47  101  171]
 [  35   55 1023]]
```

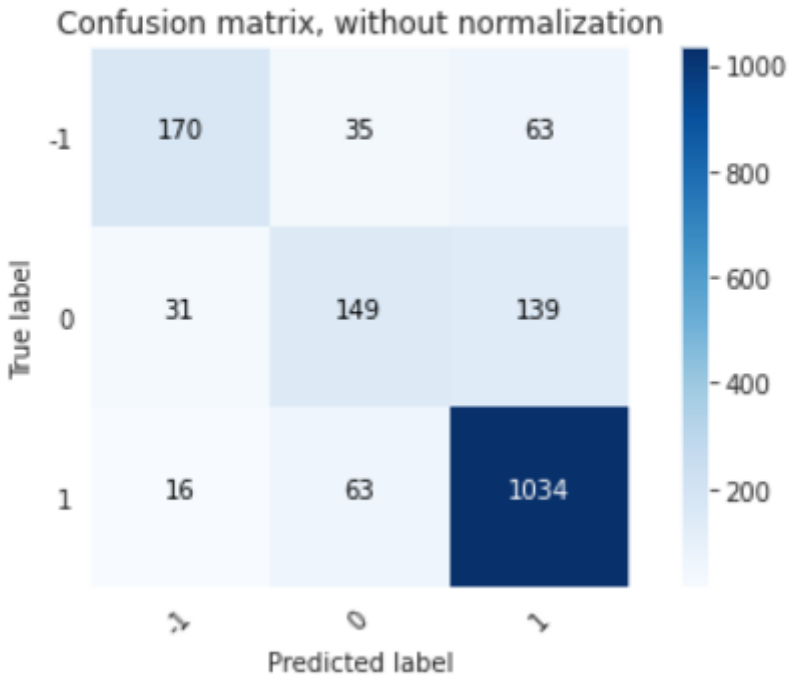


	precision	recall	f1-score	support
-1	0.63	0.53	0.58	268
0	0.51	0.32	0.39	319
1	0.80	0.92	0.86	1113
accuracy			0.74	1700
macro avg	0.65	0.59	0.61	1700
weighted avg	0.72	0.74	0.72	1700

2. Kết quả mô hình Logistic Regression

```
Model LogisticRegression
Train score:  0.8834780050022069
Test score:  0.7958823529411765
F1 score:  0.7026357740401893
```

```
Confusion matrix, without normalization
[[ 170   35   63]
 [  31  149  139]
 [  16   63 1034]]
```

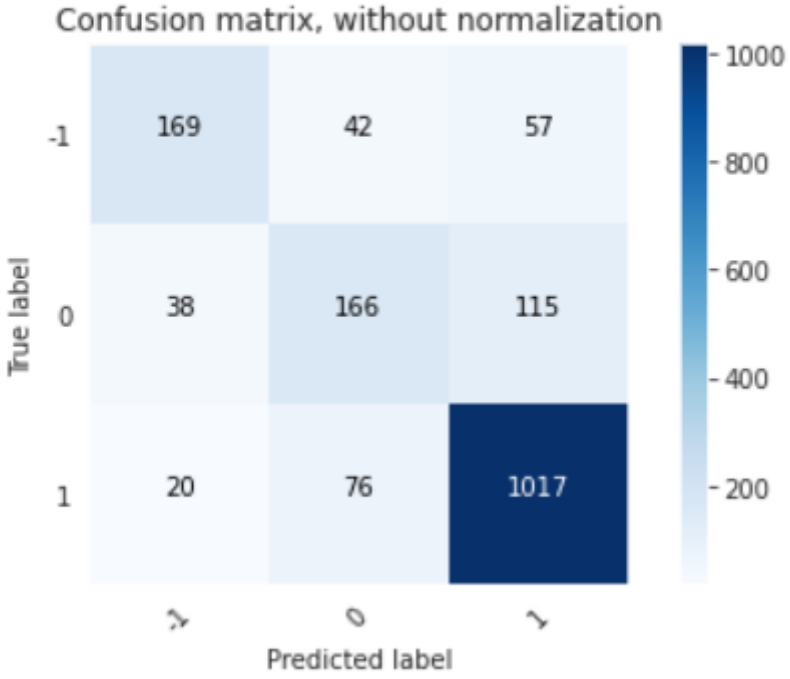


	precision	recall	f1-score	support
-1	0.78	0.63	0.70	268
0	0.60	0.47	0.53	319
1	0.84	0.93	0.88	1113
accuracy			0.80	1700
macro avg	0.74	0.68	0.70	1700
weighted avg	0.78	0.80	0.79	1700

3. Kết quả mô hình SVM

```
Model SVC
Train score:  0.9001029866117405
Test score:  0.7952941176470588
F1 score:  0.7056627366986951
```

```
Confusion matrix, without normalization
[[ 169   42   57]
 [  38  166  115]
 [  20   76 1017]]
```

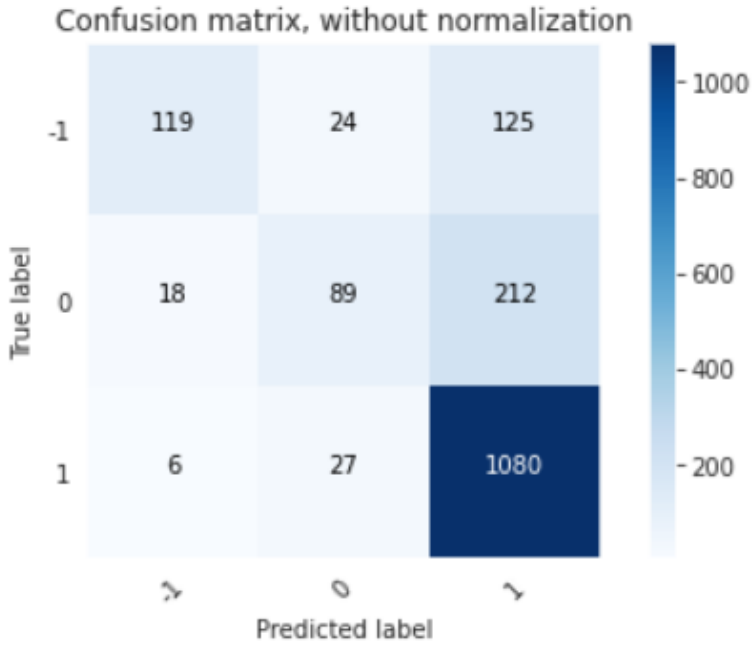


	precision	recall	f1-score	support
-1	0.74	0.63	0.68	268
0	0.58	0.52	0.55	319
1	0.86	0.91	0.88	1113
accuracy			0.80	1700
macro avg	0.73	0.69	0.71	1700
weighted avg	0.79	0.80	0.79	1700

4. Kết quả mô hình Naive Bayes

```
Model MultinomialNB
Train score:  0.8012358393408857
Test score:  0.7576470588235295
F1 score:  0.6068766435907897
```

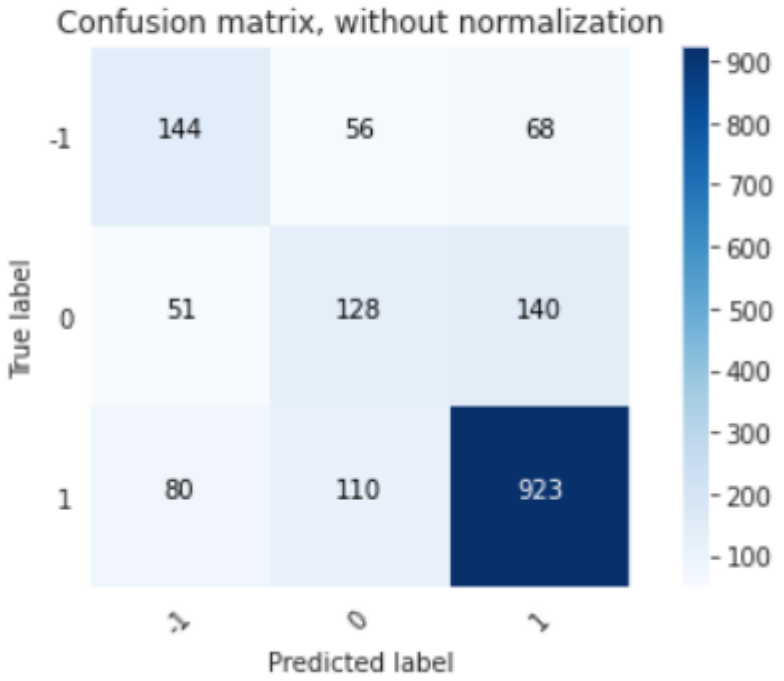
```
Confusion matrix, without normalization
[[ 119   24  125]
 [  18   89  212]
 [   6   27 1080]]
              precision    recall  f1-score   support

          -1       0.83      0.44      0.58       268
           0       0.64      0.28      0.39       319
           1       0.76      0.97      0.85      1113

    accuracy                           0.76      1700
   macro avg       0.74      0.56      0.61      1700
weighted avg       0.75      0.76      0.72      1700
```



5. Kết quả mô hình Decision Tree

```
Train score:  0.995880535530381
Test score:  0.7029411764705882
F1 score:  0.5902143857661324
```

```
Confusion matrix, without normalization
[[144  56  68]
 [ 51 128 140]
 [ 80 110 923]]
```



	precision	recall	f1-score	support
-1	0.52	0.54	0.53	268
0	0.44	0.40	0.42	319
1	0.82	0.83	0.82	1113
accuracy			0.70	1700
macro avg	0.59	0.59	0.59	1700
weighted avg	0.70	0.70	0.70	1700

Bảng so sánh kết quả chạy của các mô hình:

Kết quả	Random Forest	Logistic Regression	SVM	Naïve Bayes	Decision Tree
Recall	0.59	**0.68**	**0.69**	0.56	0.59
Precision	0.65	**0.74**	**0.73**	0.76	0.59
F1_score	0.61	**0.7**	**0.71**	0.61	0.59
Accuracy	0.74	**0.8**	**0.8**	0.75	0.7

Ta thấy rằng mô hình Random Forest và mô hình Decision Tree có kết quả rất cao trên tập train và rất tệ trên tập test chỉ (hơn 99% và 60%).Từ đó ta thấy rằng 2 mô hình trên đã bị overfitting

Ngoài ra ở cả 5 mô hình đều có sự chênh lệch lớn presicion và recall giữa các class(class 1 có kết quả tốt nhất và class 0 là thấp nhất). Nguyên nhân là data thu thập được quá mất cân bằng

Từ bảng so sánh trên nhóm chúng em thấy rằng mô hình Logistic Regression và SVM là 2 mô hình hoạt động tốt nhất. Điều này cũng khá hợp lý vì với các model ít bị ảnh hưởng bởi việc mất cân bằng dữ liệu

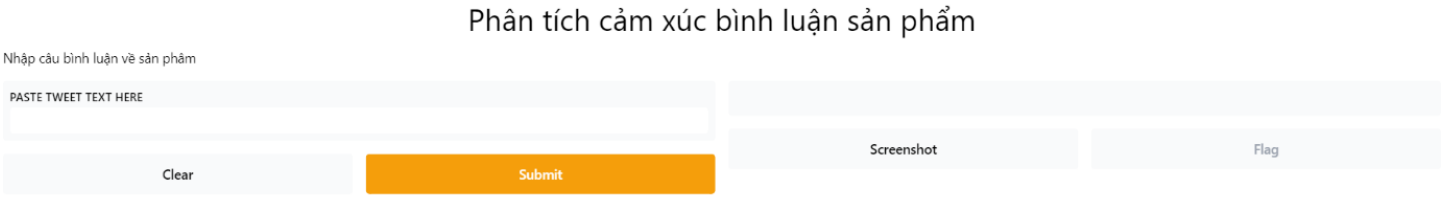
Vì thế phần demo sẽ dựa trên kết quả huấn luyện của 2 model này. Code training ở link sau: [notebook huấn luyên.]

V. Hướng phát triển và ứng dụng demo

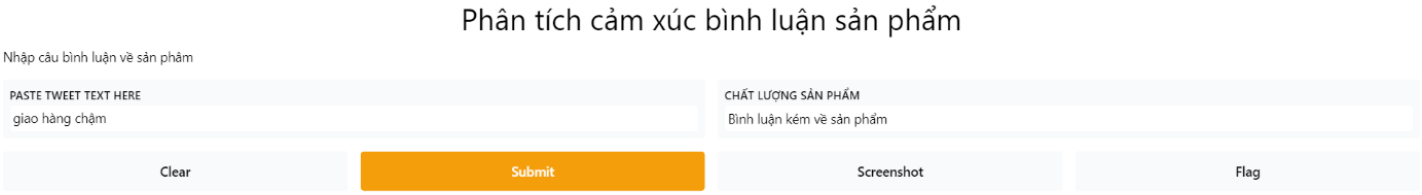
Trong tương lai nhằm tăng tính ứng dụng thực tế của mô hình, chúng em sẽ mở rộng ra thêm nhiều class hơn để có thể đánh giá mức độ hài lòng,thõa mãn của người dùng một cách tốt hơn, cũng như tăng cường thêm dữ liệu, sử dụng các model deep learning hiện đại hơn để tăng tính hiệu quả của việc huấn luyện cũng như kết quả dự đoán của mô hình.

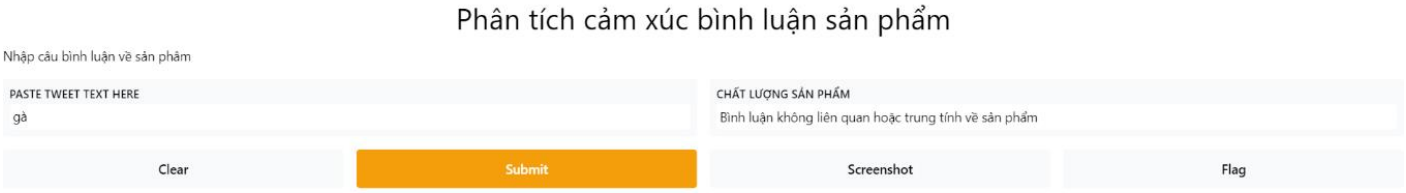
Nhóm chúng em có phát triển một ứng dụng demo đơn giản: [Link demo]

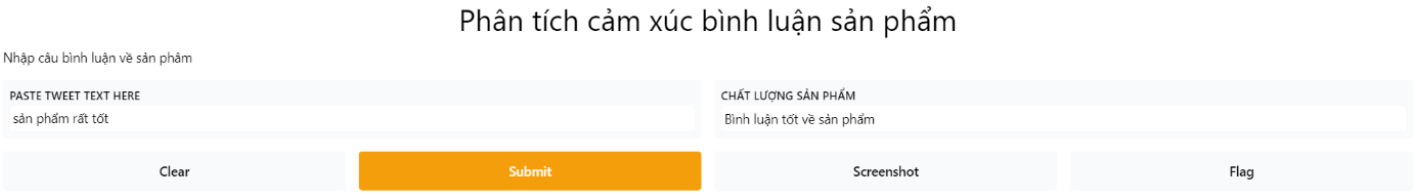
Giao diện thông thường của web:



Một số bình luận mẫu và các kết quả trả về của ứng dụng







Ngoài ra có thể xem một số kết quả thực nghiệm bình luận ở github của nhóm ở đây:

VI. Bảng phân công

Người Nhận	MSSV	Công việc được phân công	Đánh giá tỉ lệ hoàn thành
Bảo	19521242	Viết báo cáo, train model	100%
Ân	19521172	Crawl data và xây dựng demo	100%
Lộc	19520687	Label dữ liệu và format báo cáo	100%

VII. Nguồn tham khảo

- TF-IDF https://vi.wikipedia.org/wiki/Tf%E2%80%93idf
- Sentiment Analysic: https://en.wikipedia.org/wiki/Sentiment_analysis
- Nhóm bạn Trọng Khánh về flow problem solving và cách trình bày báo cáo: https://github.com/trong-khanh-1109/CS117.L22.KHCL/blob/main/Final-Report.pdf
- Computational Thinking: https://www.coursera.org/learn/compthinking/supplement/4mnoE/introduction-to-human-trafficking-case-study
- Code tham khảo: https://github.com/Long-1234kfgkl/CS114.K21/blob/master/BaoCaoCuoiKy_CS114.K21/Main.ipynb https://viblo.asia/p/phan-tich-phan-hoi-khach-hang-hieu-qua-voi-machine-learningvietnamese-sentiment-analysis-Eb85opXOK2G

Working With Text Data — scikit-learn 0.24.2 documentation
CS114.K21/Text_Classification.ipynb at master · ThuanPhong0126/CS114.K21 (github.com)