


# PHẦN 1: THÔNG TIN TÓM TẮT (18520473)

|   |   |
|---|---|
| <b>Tên đề tài (IN HOA)</b>                      | PHÁT HIỆN BÌNH LUẬN CHỐNG PHÁ, XÚC PHẠM NHÀ NƯỚC TRÊN MẠNG XÃ HỘI   |
| <b>Họ và tên (IN HOA)</b>                       | TRẦN TRUNG ANH  |
| <b>Lớp - MSSV</b>                               | CS114.K21.KHTN - 18520473   |
| <b>Ảnh</b>                                      |   |
| <b>Link Github chứa repos CS114.K21</b>         | - <a href="https://github.com/AVL1/CS114.K21.KHTN">https://github.com/AVL1/CS114.K21.KHTN</a>   |
| <b>Điểm đánh giá giữa kỳ (A B C D)</b>          | - C   |
| <b>Thành tích để tính điểm bonus</b>            | - Không có  |
| <b>Tóm tắt Bài tập quá trình</b>                | <ul style="list-style-type: none"><li>- Số lần nộp bài tập Quá trình trên Classroom:36/36</li><li>- Số lần nộp bài Thực hành trên Classroom:8/8</li><li>- Tự đánh giá (95/100):</li></ul>   |
| <b>Tóm tắt Đồ án Cuối kỳ (không quá 500 từ)</b> | <ul style="list-style-type: none"><li>- Đề tài này đề cập về việc xây dựng một công cụ nhằm phát hiện những bình luận có tính chất xúc phạm, chống phá Nhà nước Việt Nam trên mạng xã hội Facebook. Dữ liệu đầu vào của bài toán là một</li></ul> |

|                  |  |
|------------------|--|
|                  | <p>câu bình luận, từ đó phân loại câu bình luận đó với các nhãn 1 (có tính chất chống phá, xúc phạm), 0 (trung tính, không liên quan) hoặc -1 (không có tính chất chống phá, xúc phạm). Đây là bài toán phân loại văn bản Tiếng Việt, với cấu trúc, văn phạm phức tạp của Tiếng Việt, việc thu thập, tiền xử lý và trích xuất đặc trưng còn nhiều khó khăn. Trải qua thử nghiệm nhiều phương pháp xử lý cơ bản, sử dụng công cụ tách từ hiện có và các thuật toán máy học phân loại phổ biến, mô hình tốt nhất đạt được 73.47%.</p> <ul style="list-style-type: none"> <li>- Tự đánh giá (xx/100): 90/100</li> </ul> |
| <b>Link khác</b> | <ul style="list-style-type: none"> <li>- Link đến báo cáo chi tiết:<br/><a href="https://github.com/AVL1/CS114.K21.KHTN/blob/master/report.pdf">https://github.com/AVL1/CS114.K21.KHTN/blob/master/report.pdf</a></li> <li>- Link đến báo cáo slides:<br/><a href="https://github.com/AVL1/CS114.K21.KHTN/blob/master/presentation.pdf">https://github.com/AVL1/CS114.K21.KHTN/blob/master/presentation.pdf</a></li> <li>- Link đến báo cáo video:<br/><a href="https://www.youtube.com/watch?v=vc6oZZlQUUE">https://www.youtube.com/watch?v=vc6oZZlQUUE</a></li> </ul>  |

## PHẦN 2: BÁO CÁO CHI TIẾT ĐỒ ÁN CUỐI KỲ

## **I. Mô tả bài toán:**

Phát hiện bình luận chống phá, xúc phạm Nhà nước trên mạng xã hội.

Mạng xã hội là dịch vụ nối kết các thành viên cùng sở thích trên Internet lại với nhau một cách tự do, không phân biệt khoảng cách, thời gian. Đây là nơi chứa lượng thông tin khổng lồ và được cập nhật vô cùng nhanh chóng. Tuy nhiên vì thế tính xác thực, an toàn của thông tin khó được đảm bảo. Lợi dụng điều đó, một số tổ chức chống phá Nhà nước dùng mạng xã hội như một công cụ truyền tải những thông điệp, tin tức sai trái, điều này dễ khiến những người dùng thiếu thận trọng tin vào những thông tin không chính xác. Đề tài này đề cập về việc xây dựng một công cụ nhằm phát hiện những bình luận có tính chất xúc phạm, chống phá Nhà nước Việt Nam trên mạng xã hội Facebook.

**Input:** một câu bình luận

**Output:**

- 1 (Positive), nếu có tính chất xúc phạm, chống đối.
- 0 (Neutral), nếu trung tính, không liên quan.
- -1 (Negative), nếu không mang tính chất xúc phạm, chống đối.

## II. Mô tả bộ dữ liệu:

Bộ dữ liệu bình luận Tiếng Việt được thu thập từ Facebook, vì dữ liệu liên quan nhiều đến chính trị nên việc thu thập chủ yếu trên hai trang Việt Tân và Thông tin chính phủ. Bao gồm tổng cộng 4576 câu, trong đó dữ liệu được gán nhãn cụ thể: nhãn 0 có 3086 câu, nhãn 1 có 1039 câu và nhãn -1 có 451 câu.

Để thu thập bộ dữ liệu này, tôi cài đặt một công cụ sử dụng Python và Selenium. [1]

Người tham gia xây dựng bộ dữ liệu:

- Trần Trung Anh (UIT): thu thập, gán nhãn, kiểm tra, tổng hợp
- Nguyễn Hữu Trường (UIT): gán nhãn
- Nguyễn Nhật Minh (QNU): gán nhãn

Bộ dữ liệu được chia thành tập huấn luyện và tập kiểm thử với tỉ lệ 8/2.

|   | Text  | value |
|---|---|-------|
| 0 | bậy nào cali giờ đóng_cửa sao cạo móng mới mở_... | 0     |
| 1 | nói thể dân việt nam mình ngu à                   | 1     |
| 2 | đương_nhiên cảnh_sát chim dân nào đi phản đồng... | 1     |
| 3 | cảnh_sát chim ngày_xưa đi lòng tội_phạm giựt d... | 1     |
| 4 | thái lan nghèo                                    | 0     |
| 5 | ai giết bắc hà ku hợp_hiển_pháp cộng_hoà xã_hộ... | 1     |
| 6 | tao người việt nam đây tao đứng đầu thằng_cha ... | 1     |
| 7 | chính_xác dân ai giật_dây hết đừng vu thể_lực ... | 1     |
| 8 | nói thiệt ước mình đc giúp_sức đập bọn chó ngu... | 1     |
| 9 | phía bên kia chiến_tuyến người_ta công_an ngại... | 1     |

Hình 1. Dữ liệu trước tiền xử lý.

### III. Các thao tác tiền xử lý:

#### 1. Loại bỏ dấu câu và kí tự đặc biệt:

Dấu câu là những kí tự để đánh dấu cấu trúc câu hoặc tượng trưng, kí tự đặc biệt bao gồm những ký tự thường không xuất hiện trong văn viết như biểu cảm (emoji), kí tự khoa học,...

#### 2. Chuyển ký tự viết hoa thành viết thường:

Tương tự như viết tắt, hai từ giống nhau về nghĩa nhưng do viết hoa nên máy tính sẽ hiểu chúng khác nhau. Tuy nhiên có một số trường

hợp từ là tên riêng có viết hoa, có thể khác với nghĩa viết thường. Điều này rất khó giải quyết. Tuy nhiên, vấn đề này sẽ được xử lý ở bước kế tiếp.

Ví dụ: Bầu trời -> bầu trời.

### 3. Chuyển các từ viết tắt về bình thường:

Từ viết tắt là dạng rút gọn của một từ hoàn chỉnh, không khác về ý nghĩa nhưng khi xử lý trên máy tính thì chúng được hiểu như hai từ khác nhau, vì vậy ta cần thao tác này để đưa chúng về nguyên mẫu. Vì số lượng từ viết tắt rất nhiều nên bước này chỉ xử lý những từ viết tắt thường xuất hiện trong bộ dữ liệu được tổng hợp lại. Một số tên riêng cũng đã được liệt kê để trở lại dạng viết hoa sau khi bị chuyển thành viết thường từ bước 2. Bước này được thực thi sau vì nếu không chuyển thành viết thường trước, có thể dẫn đến nhập nhằng giữa viết tắt viết thường với viết tắt viết hoa.

Ví dụ: dcs -> Đảng Cộng sản

### 4. Loại bỏ stopwords:

Stopword là những từ ngữ được dùng chủ yếu để nối từ, nối câu. Chúng thường không mang thêm nhiều ý nghĩa trong câu. Danh sách stopwords trong Tiếng Việt được sử dụng là danh sách được tổng hợp

bởi nhóm nghiên cứu Association for Vietnamese Language and Speech Processing (VLSP).

Ví dụ một số stopword như *hay, lẽ, là, nhưng,...*

## 5. Tách từ:

Tách từ, hay còn gọi là tokenize, là quá trình gom các chữ kề nhau lại thành một từ có ý nghĩa hoàn chỉnh. Vì một từ trong Tiếng Việt có thể được ghép từ nhiều tiếng, mỗi tiếng có thể có hoặc không có ý nghĩa riêng. Quá trình tách được thực hiện qua việc sử dụng công cụ pyvi.

Chẳng hạn, từ *trông trọt* nếu đứng riêng lẻ thì từ *trọt* sẽ không có ý nghĩa. Nên ta cần chuyển *trông trọt* thành *trông\_trọt* như một từ.

|   | Text  | value |
|---|---|-------|
| 0 | bậy nào cali giờ đóng_cửa sao cạo móng mới mở_... | 0     |
| 1 | nói thể dân việt nam mình ngu à                   | 1     |
| 2 | đương_nhiên cảnh_sát chim dân nào đi phản đồng... | 1     |
| 3 | cảnh_sát chim ngày_xưa đi lòng tội_phạm giựt d... | 1     |
| 4 | thái lan nghèo                                    | 0     |
| 5 | ai giết bắc hà ku hợp_hiển_pháp cộng_hoà xã_hộ... | 1     |
| 6 | tao người việt nam đây tao đứng đầu thẳng_cha ... | 1     |
| 7 | chính_xác dân ai giết_dây hết đừng vu thể_lực ... | 1     |
| 8 | nói thiệt ước mình đc giúp_sức đập bọn chó ngu... | 1     |
| 9 | phía bên kia chiến_tuyến người_ta công_an ngại... | 1     |

Hình 2. Dữ liệu sau tiền xử lý.

## **IV. Phương pháp trích xuất đặc trưng:**

### **1. CountVectorize:**

Xây dựng một bộ từ điển gồm các từ khác nhau, tổng hợp từ bộ dữ liệu kèm với số lần từ đó xuất hiện trong toàn bộ dữ liệu.

Sau đó, trong mỗi câu, với mỗi từ trong từ điển, ta đếm số lần xuất hiện của từ đó trong câu.

Từ đó, ta có mỗi vector đặc trưng với độ dài bằng độ dài của từ điển.

### **2. TF-IDFVectorize:**

Phương pháp phổ biến là sử dụng một phương pháp thống kê có tên là TF-IDF, giá trị TF-IDF của một từ là một con số 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 nằm trong một tập hợp các văn bản.

Giá trị TF-IDF được xác định:

$$\text{TF-IDF value} = \text{TF} * \log(\text{IDF}) \text{ với}$$



- Term Frequency: tần số xuất hiện của một từ trong một câu.
- Inverse Document Frequency: nghịch đảo tần số của một từ trong toàn bộ các câu.

## **V. Lựa chọn thuật toán máy học:**

Đây là dạng bài toán phân loại văn bản nên hai thuật toán phổ biến thường được sử dụng là Multinomial Naive Bayes và Support Vector Machine.

### **1. Multinomial Naive Bayes:**

Đây là một trong những phương pháp được xây dựng dựa trên định lý Bayes với giả định các đặc trưng hoàn toàn độc lập với nhau. MNB được chủ yếu dùng cho bài toán phân loại nhiều nhãn.

Được xác định từ công thức Bayes:

The diagram shows the Naive Bayes formula with arrows pointing from labels to the corresponding parts of the equation:

- Likelihood** points to  $P(x|c)$
- Class Prior Probability** points to  $P(c)$
- Posterior Probability** points to  $P(c|x)$
- Predictor Prior Probability** points to  $P(x)$

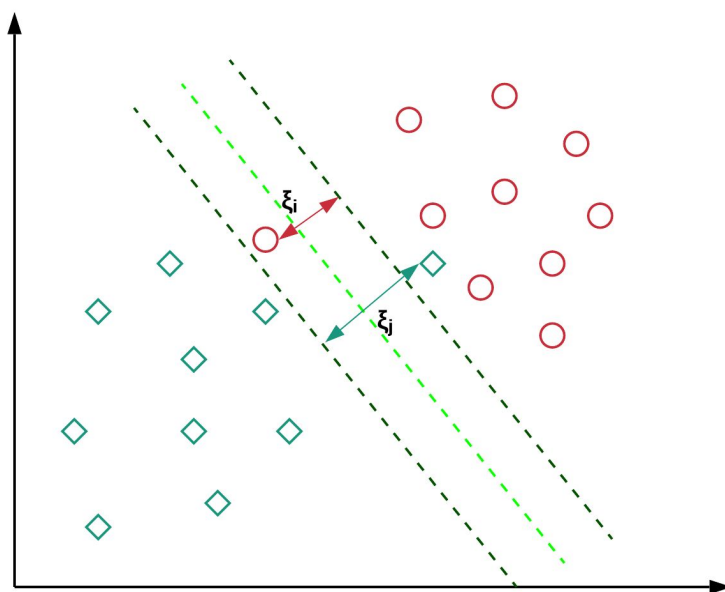
$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c)$$

Hình 3. Naive Bayes rule.

## 2. Support Vector Machine:

Ý tưởng của thuật toán SVM là đi tìm các siêu mặt phẳng trong không gian N chiều sao cho chúng có thể phân biệt nhóm các điểm dữ liệu với nhau. Khoảng cách giữa các điểm dữ liệu của các nhãn khác nhau đến mặt phẳng càng cân bằng thì kết quả dự đoán càng tốt.



Hình 4. Support Vector Machine.

## VI. Kết quả:

Quá trình kiểm thử được thực hiện trên tập kiểm thử gồm 915 câu. Kết quả accuracy của mỗi mô hình được thể hiện ở bảng dưới:

|           | LSVC     | MNB      |
|-----------|----------|----------|
| <b>CV</b> | 0.728166 | 0.713974 |
| <b>TV</b> | 0.734716 | 0.658297 |

Hình 5. Kết quả thực nghiệm.

Trong đó, LSVC là Linear SVC, MNB là Multinomial Naive Bayes, CV là Count Vectorize và TV là TF-IDF Vectorize.

Thực hiện Grid Search cho mô hình LSVC và TV để tối ưu tham số C (regularization), ta tìm được giá trị C phù hợp là 10,1.

Kết quả với mô hình LSVC và TV với tham số C là 10,1.

|              | precision | recall | f1-score | support |
|--------------|-----------|--------|----------|---------|
| -1           | 0.59      | 0.40   | 0.48     | 96      |
| 0            | 0.76      | 0.85   | 0.80     | 595     |
| 1            | 0.58      | 0.48   | 0.52     | 225     |
| accuracy     |           |        | 0.71     | 916     |
| macro avg    | 0.64      | 0.57   | 0.60     | 916     |
| weighted avg | 0.70      | 0.71   | 0.70     | 916     |

Hình 6. Kết quả mô hình với tham số tối ưu.

## VII. Nhận xét, đánh giá:

Mô hình phân loại đúng một số câu bình luận phổ biến.

```
[87] inp = [
    'Chỉ có bọn quan tham cộng sản xhcn vô pháp vô thiên mới hành xử như bọn mafia, ăn chặn trên xương máu của người dân!',
    'Nhạn nhân những chuyện tương tự thế này trên khắp Việt Nam.',
    'Tuyệt vời, cảm ơn Đảng, chính phủ luôn quan tâm đến người dân dù ở trong hay ngoài nước.'
]
inp = preprocess(inp)
inp_vector = tv.transform(inp)
pred = bestLinearSVC.predict(inp_vector)
print('Result: ', pred)
```

Result: [ 1 0 -1]

Hình 7. Kết quả khi sử dụng mô hình với dữ liệu mới.

Phân loại dữ liệu nhập mới từ người dùng chưa thực sự chính xác.

Các mô hình cho kết quả khá thấp vì các nguyên nhân:

- Bộ dữ liệu mất cân bằng, tiền xử lý không hoàn toàn chính xác cho từng câu, quan trọng nhất là vấn đề sai chính tả, văn phạm.
- Công cụ tách từ chưa tốt cho dữ liệu Tiếng Việt.
- Số lượng dữ liệu còn hạn chế, các bình luận có thể không đủ ý.
- Có trường hợp một số câu gán nhãn chưa đúng.
- Chưa xử lý được trường hợp overfitting do một từ xuất hiện quá nhiều.

## **VIII. Hướng phát triển thêm:**

Xây dựng, phát triển bộ dữ liệu:

- Tăng số lượng, cân bằng các nhãn.
- Tiền xử lý các câu sai chính tả, tên riêng.
- Thu thập từ nhiều nguồn khác nhau.
- Đảm bảo đồng bộ, chính xác trong quá trình gán nhãn.

Từ bộ dữ liệu hoàn chỉnh hơn, ứng dụng các phương pháp học sâu (Deep Learning) vào trích xuất đặc trưng.

Cải thiện việc tối ưu trên nhiều tham số.

