

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM  
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG  
KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KỲ MÔN  
KHAI THÁC DỮ LIỆU VÀ KHAI PHÁ TRI THỨC**

**NHẬN BIẾT CẢM XÚC ĐỐI VỚI VĂN BẢN  
TRÊN MẠNG XÃ HỘI VIỆT NAM, SỬ DỤNG  
TẬP DỮ LIỆU UIT-VSMEC VÀO TẬP HUẤN  
MÔ HÌNH VÀ BÁO CÁO HIỆU SUẤT**

*Giảng viên giảng dạy:* **TS. Lê Cung Trường**

*Người thực hiện:* **TÔ VĨNH KHANG - 51800408**

**BÙI QUANG KHẢI - 51800785**

**Khoá : 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM**  
**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**  
**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO CUỐI KỲ MÔN**  
**KHAİ THÁC DỮ LIỆU VÀ KHAİ PHÁ TRI THỨC**

**NHẬN BIẾT CẢM XÚC ĐỐI VỚI VĂN BẢN**  
**TRÊN MẠNG XÃ HỘI VIỆT NAM, SỬ DỤNG**  
**TẬP DỮ LIỆU UIT-VSMEC VÀO TẬP HUẤN**  
**MÔ HÌNH VÀ BÁO CÁO HIỆU SUẤT**

*Giảng viên giảng dạy:* **TS. Lê Cung Tường**

*Người thực hiện:* **TÔ VĨNH KHANG - 51800408**

**BÙI QUANG KHẢI - 51800785**

**Khoá : 22**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

## LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn Khoa Công nghệ thông tin và Trường Đại học Tôn Đức Thắng đã tạo điều kiện cho chúng em được học tập trong suốt thời gian qua. Chân thành cảm ơn Thầy Lê Cung Tường đã giúp chúng em có thêm kiến thức về khai thác dữ liệu và khai phá tri thức. Tìm hiểu thêm được nhiều phương pháp trong việc phân tích dữ liệu thực tế hiện nay.

Trong quá trình thực hiện bài báo cáo này nhóm vẫn khó tránh khỏi những sai sót không mong muốn, kính mong thầy có thể góp ý và giúp đỡ chúng em. Nhóm xin chân thành cảm ơn thầy.

## **BÁO CÁO NÀY ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Chúng em xin cam đoan đây là sản phẩm của riêng chúng em được sự hướng dẫn của thầy Lê Cung Tường. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính chúng em thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong bài báo cáo này còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào chúng em xin hoàn toàn chịu trách nhiệm về nội dung bài báo cáo của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do chúng em gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày 15 tháng 04 năm 2021*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Tô Vĩnh Khang*

*Bùi Quang Khải*

## PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

### Phần xác nhận của GV hướng dẫn

---

---

---

---

---

---

---

Tp. Hồ Chí Minh, ngày    tháng    năm  
(ký và ghi họ tên)

### Phần đánh giá của GV chấm bài

---

---

---

---

---

---

---

Tp. Hồ Chí Minh, ngày    tháng    năm  
(ký và ghi họ tên)

## TÓM TẮT

Ngày nay, công nghệ thông tin đã và đang đóng vai trò quan trọng trong xã hội. Nó giúp con người làm việc thoải mái hơn, dễ dàng hơn. Thông qua những bước phân tích dữ liệu, ta có thể đánh giá, dự đoán những vấn đề trong cuộc sống. Từ đó, quản lý và tổ chức công việc đạt hiệu quả cao. Việc ứng dụng khai thác dữ liệu sẽ giúp khai phá tri thức cho nhân loại, cho mọi lĩnh vực của đời sống xã hội.

Ở bài báo cáo này, chúng em đã thu thập dữ liệu từ 3 video youtube với số lượng 100 bình luận đầu tiên. Qua nhiều công đoạn tiền xử lý dữ liệu để giúp tập dữ liệu thu được sạch. Tiếp theo đó là sử dụng tập dữ liệu được cung cấp từ UIT-VSMEC để tập huấn dữ liệu, tính toán các số liệu hiệu suất như độ chính xác, độ F1. Sau đó, dùng chính tập dữ liệu đã tập huấn để đưa vào 3 tập dữ liệu đã thu thập được từ 3 video youtube. Tính toán các số liệu hiệu suất tương ứng và so sánh, lập biểu đồ.

Các thư viện chính mà chúng em sử dụng gồm: Thư viện selenium phục vụ cho việc thu thập dữ liệu từ video youtube; Thư viện pandas và scikit-learn sẽ phục vụ trong việc xử lý dữ liệu.

# MỤC LỤC

LỜI CẢM ƠN.....	3
TÓM TẮT.....	6
MỤC LỤC.....	7
DANH MỤC CÁC BẢNG, HÌNH.....	9
DANH MỤC HÌNH.....	9
DANH MỤC BẢNG.....	9
CHƯƠNG I: GIỚI THIỆU CHUNG.....	10
1.1 Giới thiệu đề tài.....	10
1.2 Tập dữ liệu UIT-VSMEC.....	10
1.3 Mô hình hóa.....	12
1.4 Thư viện sử dụng chính.....	12
CHƯƠNG II: CƠ SỞ LÝ THUYẾT.....	13
2.1 Định nghĩa cảm xúc.....	13
2.2 Một số khái niệm cơ bản.....	14
2.2.1 Cào dữ liệu (Crawling Data).....	14
2.2.2 Tiền xử lý dữ liệu (Data Preprocessing).....	14
2.2.3 Học cây quyết định (Decision Tree Learning).....	14
2.2.4 Số liệu hiệu suất (Performance metrics).....	15
2.2.4.1 Accuracy Score (AS).....	15
2.2.4.2 Precision Score (PS).....	15
2.2.4.3 F1 Score (FS).....	15
CHƯƠNG III: HIỆN THỰC BẰNG CODE PYTHON.....	16
3.1 Sử dụng thư viện.....	16

3.2 Thu thập dữ liệu từ 3 video youtube.....	16
3.2.1 Cào dữ liệu (Crawling data).....	16
3.2.2 Gán nhãn cảm xúc (Label emotions).....	19
3.2.3 Một số đóng góp (Contributions).....	22
3.3 Nhận biết cảm xúc đối với văn bản trên mạng xã hội Việt Nam.....	25
3.3.1 Huấn luyện mô hình từ tập dữ liệu tập huấn và thẩm định của UIT- VSMEC.....	25
3.3.2 Số liệu hiệu suất.....	28
3.4 Áp dụng mô hình đã tập huấn từ tập dữ liệu UIT-VSMEC.....	28
3.4.1 Áp dụng mô hình đã tập huấn cho 3 tập dữ liệu từ video youtube.....	28
3.4.2 Số liệu hiệu suất.....	30
CHƯƠNG IV: TỔNG KẾT.....	33
TÀI LIỆU THAM KHẢO.....	34



# DANH MỤC CÁC BẢNG, HÌNH

## DANH MỤC HÌNH

<i>Hình 1. Mô hình hóa về các hoạt động được triển khai.....</i>	<i>12</i>
<i>Hình 2. Các biểu đồ của 3 tập dữ liệu thu được từ video kênh youtube Khoa Pug.....</i>	<i>24</i>
<i>Hình 3. Một đoạn trích từ tập dữ liệu UIT-VSMEC sau quá trình tập huấn.....</i>	<i>27</i>
<i>Hình 4. Một đoạn trích từ tập dữ liệu của Video Youtube 1 sau quá trình tập huấn.....</i>	<i>29</i>
<i>Hình 5. Một đoạn trích từ tập dữ liệu của Video Youtube 2 sau quá trình tập huấn.....</i>	<i>30</i>
<i>Hình 6. Một đoạn trích từ tập dữ liệu của Video Youtube 3 sau quá trình tập huấn.....</i>	<i>30</i>

## DANH MỤC BẢNG

<i>Bảng 1. Bảng thống kê số lượng nhãn cảm xúc trong tập dữ liệu UIT-VSMEC.....</i>	<i>11</i>
<i>Bảng 2. Bảng các thư viện chính được sử dụng.....</i>	<i>12</i>
<i>Bảng 3. Bảng định nghĩa cảm xúc.....</i>	<i>14</i>

# CHƯƠNG I: GIỚI THIỆU CHUNG

## 1.1 Giới thiệu đề tài

Ngày nay, công nghệ thông tin đã và đang đóng vai trò quan trọng trong xã hội. Nó giúp con người làm việc thoải mái hơn, dễ dàng hơn. Thông qua những bước phân tích dữ liệu, ta có thể đánh giá, dự đoán những vấn đề trong cuộc sống. Từ đó, quản lý và tổ chức công việc đạt hiệu quả cao. Việc ứng dụng khai thác dữ liệu sẽ giúp khai phá tri thức cho nhân loại, cho mọi lĩnh vực của đời sống xã hội.

Với đề tài cuối kì “Nhận biết cảm xúc đối với văn bản trên mạng xã hội Việt Nam sử dụng tập dữ liệu UIT-VSMEC tập huấn mô hình và báo cáo hiệu suất”, chúng em xin trình bày về các quá trình thực hiện đồ án gồm: Thu thập dữ liệu là những bình luận từ 3 video trên mạng xã hội Youtube, gán nhãn cảm xúc với 7 loại (thích thú, chán ghét, sợ hãi, tức giận, buồn bã, ngạc nhiên và khác), tiền xử lý dữ liệu văn bản, sử dụng tập dữ liệu được cung cấp từ UIT-VSMEC để tập huấn dữ liệu và áp dụng mô hình được tập huấn cho 3 tập dữ liệu bình luận thu được từ video. Cuối cùng là tính toán các số liệu hiệu suất như độ chính xác, độ F1,... Bên cạnh đó, chúng em còn sử dụng một số biểu đồ để thống kê dữ liệu thu nhận được.

Các thư viện chính được sử dụng trong bài đồ án này gồm: Thư viện Selenium phục vụ cho việc thu thập dữ liệu từ video youtube; Thư viện Pandas, Scikit-learn và Underthesea sẽ phục vụ trong việc xử lý dữ liệu.

## 1.2 Tập dữ liệu UIT-VSMEC

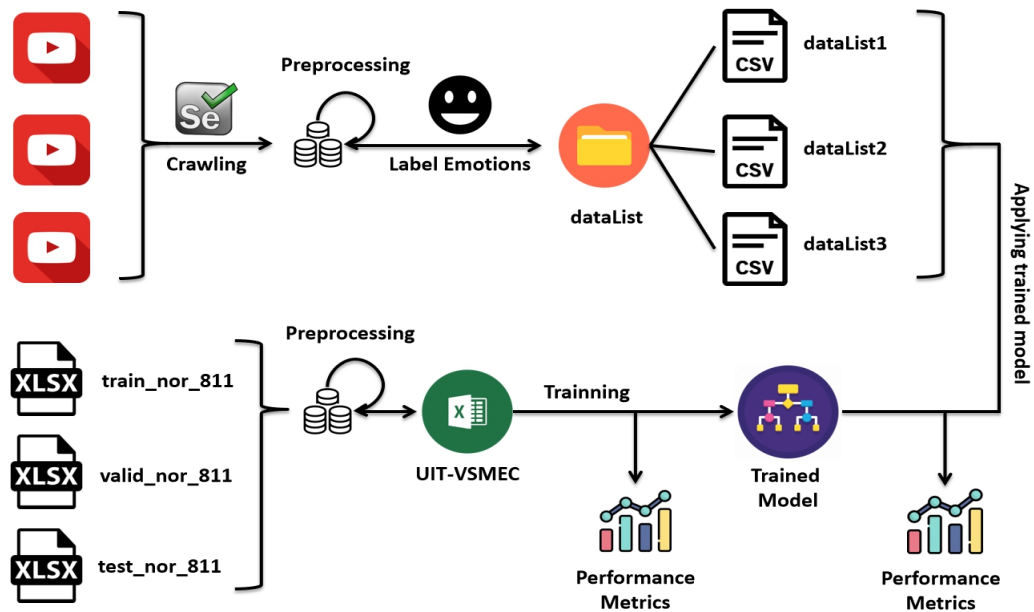
Tập dữ liệu UIT-VSMEC (University of Information Technology - Vietnamese Social Media Emotion Corpus) là tập dữ liệu được thu thập từ 6.927 câu được chú thích bởi con người với 6 nhãn cảm xúc. Nó góp phần rất nhiều vào việc nghiên cứu nhận dạng cảm xúc bằng tiếng Việt trong Xử lý ngôn ngữ tự nhiên.

- + Tập dữ liệu tập huấn (5548 câu ~80%): *train\_nor\_811.xlsx*
- + Tập dữ liệu thẩm định (686 câu ~10%): *valid\_nor\_811.xlsx*
- + Tập dữ liệu kiểm thử (693 câu ~10%): *test\_nor\_811.xlsx*

<b>Tên cảm xúc</b>	<b>Số lượng câu</b>	<b>Tỉ số phần trăm (%)</b>
Thích thú	1965	28.36
Chán ghét	1338	19.31
Sợ hãi	1149	16.59
Tức giận	0480	06.92
Buồn bã	0395	05.70
Ngạc nhiên	0309	04.46
Khác	1291	18.66





***Bảng 1. Bảng thống kê số lượng nhãn cảm xúc trong tập dữ liệu UIT-VSMEC***

### 1.3 Mô hình hóa



Hình 1. Mô hình hóa về các hoạt động được triển khai

### 1.4 Thư viện sử dụng chính

Tên thư viện	Hình ảnh logo	Phiên bản
selenium		3.141.0
scikit-learn		0.23.2
pandas		1.1.3
underthesea		1.3.1

Bảng 2. Bảng các thư viện chính được sử dụng

# CHƯƠNG II: CƠ SỞ LÝ THUYẾT

## 2.1 Định nghĩa cảm xúc

Tên cảm xúc	Mô tả
Thích thú	Những bình luận thể hiện sự thích thú, phấn khích. Nó chứa đựng cả sự yên bình và ngây ngất. Khi họ nhìn thấy lòng tốt và lòng trắc ẩn, trải nghiệm thoải mái và mãn nguyện hoặc thậm chí là tận hưởng những giai điệu sai lầm của người khác với niềm tự hào vui sướng về những thành tích hoặc trải nghiệm về một cái gì đó rất đẹp, tuyệt vời.
Chán ghét	Những bình luận thể hiện sự không thích và ghê tởm. Cảm giác như muốn tránh điều gì đó ghê tởm hoặc ác cảm, phản ứng với một mùi vị, mùi, sự vật hoặc ý tưởng xấu, sự ghê tởm, hận thù.
Sợ hãi	Những bình luận thể hiện sự lo lắng, khiếp sợ. Cảm giác như dự đoán về khả năng gặp nguy hiểm, lo lắng, sợ hãi, hoảng sợ, sự việc ghê tởm đến sốc.
Tức giận	Những bình luận thể hiện sự khó chịu và giận dữ. Khi bị phỉ nhổ toái, tranh luận mạnh mẽ đến cay đắng, tuyệt vọng khi lặp đi lặp lại một việc mà muốn vượt qua nhưng không thể.
Buồn bã	Những lời bình chứa đựng cả sự hụt hẫng và tuyệt vọng. Cảm giác chán nản, quẫn trí, bất lực, đau khổ., một sự việc gây mất mát hoặc đau buồn.
Ngạc nhiên	Những bình luận thể hiện cảm xúc của bất ngờ. Chứng kiến một sự việc khó tin đến sốc. Cảm xúc ngẩn ngui trong tất cả các cung bậc

	cảm xúc, chỉ diễn ra trong vài giây và sau đó mất đi khi đã hiểu những gì đang xảy ra và nó có thể sẽ trở thành những cảm giác khác như sợ hãi, tức giận, nhẹ nhõm...
Khác	Những bình luận không thể hiện cảm xúc nào hoặc không thể hình dung hay miêu tả được.

***Bảng 3. Bảng định nghĩa cảm xúc***

## **2.2 Một số khái niệm cơ bản**

### **2.2.1 Cào dữ liệu (Crawling Data)**

Là quá trình thu thập thông tin từ đường đường dẫn bất kỳ được cung cấp. Nó tiến hành phân tích mã nguồn HTML để đọc dữ liệu và lọc ra những thông tin theo yêu cầu của người dùng.

### **2.2.2 Tiền xử lý dữ liệu (Data Preprocessing)**

Là loại bỏ dữ liệu không mong muốn bằng việc làm sạch dữ liệu, giúp cho dữ liệu thu được có giá trị hơn sau giai đoạn tiền xử lý để thao tác dữ liệu sau này trong quá trình khai thác dữ liệu. Do đó, tính đại diện và chất lượng của dữ liệu là đầu tiên và quan trọng nhất trước khi chạy bất kỳ phân tích nào. Khi có nhiều thông tin không liên quan và dư thừa hiện tại hoặc dữ liệu nhiễu và không đáng tin cậy thì việc khai phá tri thức sẽ khó khăn hơn. Bên cạnh đó, tiền xử lý dữ liệu có thể ảnh hưởng đến cách diễn giải kết quả của quá trình xử lý dữ liệu cuối cùng.

### **2.2.3 Học cây quyết định (Decision Tree Learning)**

Là phương pháp sử dụng cây quyết định như một mô hình dự đoán. Từ các quan sát về một tập dữ liệu đến kết luận về giá trị mục tiêu của tập dữ liệu đó. Lúc này, lá đại diện cho nhãn lớp và cành biểu thị các liên từ của các tính năng dẫn đến các nhãn lớp đó.

Tuy nhiên, một thay đổi nhỏ trong dữ liệu tập huấn có thể dẫn đến sự thay đổi lớn trong cây, có thể tạo cây quá phức tạp mà không tổng quát hóa tốt (overfitting).

#### 2.2.4 Số liệu hiệu suất (Performance metrics)

*Khẳng định đúng (TP): Là những giá trị dương được dự đoán chính xác. Giá trị của lớp thực tế là có và giá trị của lớp được dự đoán cũng là có.*

*Phủ định đúng (TN): Là các giá trị âm được dự đoán chính xác. Giá trị của lớp thực tế là không và giá trị của lớp được dự đoán cũng là không.*

*Khẳng định sai (FP): Khi lớp thực tế là không và lớp dự đoán là có.*

*Phủ định sai (FN): Khi lớp thực tế là có nhưng lớp dự đoán là không.*

*Thu hồi (RC): Là tỉ lệ giữa các quan sát tích cực được dự đoán chính xác so với tất cả các quan sát trong lớp thực tế.  $RC = (TP/TP) + FN$*

##### 2.2.4.1 Accuracy Score (AS)

Là thước đo hiệu suất trực quan nhất và nó chỉ đơn giản là tỷ lệ giữa quan sát được dự đoán chính xác trên tổng số quan sát. Nó là một thước đo tuyệt vời nhưng chỉ khi tập dữ liệu đối xứng trong đó các FN và FP gần như giống nhau. Do đó, cần xem xét các thông số khác để đánh giá hiệu suất của mô hình.

$$AS = (TN/TP) + (TN+TP) + (FN+FP)$$

##### 2.2.4.2 Precision Score (PS)

Là tỷ lệ của các quan sát tích cực được dự đoán chính xác trên tổng số các quan sát tích cực được dự đoán.

$$PS = (TP/TP) + FP$$

##### 2.2.4.3 F1 Score (FS)

Là trọng số trung bình của PS và RC. Nó khó để hiểu nhưng thường hữu ích hơn AS. Đặc biệt cho những phân bố lớp không đồng đều. AS hoạt động tốt nhất nếu FN và FP có chi phí tương tự. Nếu FN và FP rất khác nhau, thì tốt hơn nên xem xét cả PS và RC.

$$FS = 2 * (RC*PS) / (RC+PS)$$

## CHƯƠNG III: HIỆN THỰC BẰNG CODE PYTHON

Yêu cầu: Python( $\geq 3.6$ ), NumPy( $\geq 1.13.3$ ), SciPy( $\geq 0.19.1$ ), Joblib( $\geq 0.11$ ), Threadpoolctl( $\geq 2.0.0$ ), Torch(1.5.0), Torchvision(0.7.0).

### 3.1 Sử dụng thư viện

```
import os
import time
import pandas as pd
from selenium.webdriver import Chrome
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import WebDriverWait as WDW
from selenium.webdriver.support import expected_conditions as EC
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import f1_score, accuracy_score, precision_score
from sklearn.feature_extraction.text import TfidfVectorizer
from underthesea import word_tokenize
from xlrd import open_workbook
import matplotlib.pyplot as plt
%matplotlib inline
```

### 3.2 Thu thập dữ liệu từ 3 video youtube

#### 3.2.1 Cào dữ liệu (Crawling data)

Trước tiên, tạo một hàm ChromeDriver(url,t) với “url” là đường dẫn video youtube cần crawl, “t” là thời gian chờ cho mỗi lần WebDriver của Selenium sử dụng để điều khiển Chrome scroll xuống.



```

def ChromeDriver(url,t):
    data = []
    with Chrome(executable_path = r'./chromedriver') as ChD:
        wait = WDW(ChD,t)
        ChD.get(url)
        for item in range(101):
            wait.until(EC.visibility_of_element_located((By.TAG_NAME, "body"))).send_keys(Keys.END)
            time.sleep(t)
            for comment in wait.until(EC.presence_of_all_elements_located
            ((By.CSS_SELECTOR, "#content"))):
                if(len(str(comment.text))>10):
                    data.append(comment.text)
            if(len(data)==103):
                break
    return data

```

Khởi tạo biến “data” được dùng để lưu trữ những dữ liệu sau khi thu thập được. Gọi Chrome thông qua đường dẫn đến file thực thi chromedriver. Ở đây là “C:\Users\MSI GAMING\chromedriver.exe”. Biến “wait” cung cấp một số độ trễ giữa các hành động được thực hiện - chủ yếu là định vị một phần tử hoặc bất kỳ hoạt động nào khác với phần tử. Chromedriver sẽ điều khiển “url” được cung cấp. Sử dụng vòng lặp for để lấy các dữ liệu “item” với vị trí có tên thẻ là “body”. Sau đó quét để lấy các bình luận với vị trí có id là “content”. Hàm if đầu tiên được dùng để loại bỏ những câu bình luận ngắn, không mang ý nghĩa như phút:giây của video, spam kí tự,.. Hàm if sau kiểm tra độ dài “data”, nếu nó đã đủ 103 thì dừng và xuất ra kết quả. Sở dĩ ở đây chọn con số 103 là vì khi lấy dữ liệu từ id “content” bao gồm 100 bình

luyện của người xem + 3 thông tin dư thừa (tiêu đề, mô tả và bình luận đầu tiên của Khoa Pug - sở thích riêng của anh ấy).

```
def getData_Video(url,t):  
    data = ChromeDriver(url,t)  
    while(len(data)==103):  
        for i in range(3):  
            data.pop(0)  
        print("Success!! Crawled 100 comments from Video [" +  
            url.split("v=")[1] + "]!")  
        return data  
    print("Failed!! Please try again.")  
    return data
```

Hàm `getData_Video(url,t)` này được dùng để xử lý loại bỏ 3 thông tin dư thừa đầu tiên như mình đã giải thích ở trên thông qua hàm `pop()`. Khi xác nhận rằng tập dữ liệu thu được là đúng 100 bình luận thì thông báo thành công. Ngược lại, thông báo thất bại.

```
url1 = "https://www.youtube.com/watch?v=hassqXTvsXM"  
url2 = "https://www.youtube.com/watch?v=RG-dXrbRNuw"  
url3 = "https://www.youtube.com/watch?v=ipSpPzFqNH0"
```

```
storageDir = "dataList/"  
if not os.path.exists(storageDir):  
    os.makedirs(storageDir)
```

Chọn 3 video youtube từ kênh Khoa Pug tương ứng lần lượt với url1, url2, url3. Tiến thành tạo folder mang tên “dataList”. Kiểm tra xem folder đã tồn tại hay chưa. Nếu chưa tồn tại thì tiến hành tạo folder đó. Thực hiện cào dữ liệu và lưu vào 3 biến data1, data2, data3 tương ứng.

```
data1 = getData_Video(url1,0.035)
```

```
data2 = getData_Video(url2,0.040)
```

```
data3 = getData_Video(url3,0.025)
```

**Kết quả:**

```
Success!! Crawled 100 comments from Video [hassqXTvsXM]!
```

```
Success!! Crawled 100 comments from Video [RG-dXrbRNuw]!
```

```
Success!! Crawled 100 comments from Video [ipSpPzFqNH0]!
```

### 3.2.2 Gán nhãn cảm xúc (Label emotions)

Thực hiện việc gán nhãn cảm xúc thủ công cho từng câu bình luận. Ở đây, nhóm đã đọc kỹ hết 300 bình luận cùng với việc dựa theo bảng định nghĩa cảm xúc (Bảng 3) để gán nhãn thủ công trực tiếp trên code để tiện cho việc sử dụng sau này.

```
DG = "Disgust"; EJ = "Enjoyment"; AG = "Anger"; SP = "Surprise";
```

```
SN = "Sadness"; FE = "Fear"; OT = "Other"
```

```
emotion1 = [
```

```
OT,EJ,EJ,OT,OT,SN,SN,OT,EJ,EJ,
```

```
SP,EJ,EJ,OT,OT,OT,EJ,OT,EJ,EJ,
```

```
SP,EJ,EJ,OT,SP,OT,EJ,EJ,EJ,OT,
```

```
EJ,EJ,EJ,OT,OT,OT,SN,OT,EJ,DG,
```

```
EJ,EJ,EJ,EJ,EJ,EJ,OT,OT,EJ,EJ,
```

```
EJ,FE,EJ,OT,OT,OT,EJ,OT,EJ,OT,
```

```
OT,SP,SP,OT,OT,EJ,EJ,OT,EJ,EJ,
```

```
DG,SP,AG,OT,OT,EJ,OT,SN,EJ,EJ,
```

OT,EJ,EJ,EJ,OT,EJ,OT,FE,EJ,EJ,  
EJ,OT,OT,SP,OT,OT,DG,OT,EJ,OT  
]

emotion2 = [  
OT,EJ,SN,EJ,SP,OT,OT,FE,OT,EJ,  
SP,EJ,EJ,AG,OT,OT,SN,EJ,OT,OT,  
OT,EJ,SN,EJ,EJ,DG,OT,OT,EJ,EJ,  
EJ,EJ,EJ,SN,OT,OT,EJ,EJ,EJ,OT,  
SN,OT,EJ,EJ,EJ,EJ,OT,EJ,OT,OT,  
OT,OT,EJ,EJ,OT,OT,EJ,OT,OT,OT,  
OT,EJ,EJ,EJ,EJ,EJ,OT,OT,EJ,EJ,  
EJ,OT,OT,OT,OT,SN,OT,EJ,EJ,EJ,  
OT,OT,EJ,OT,OT,EJ,EJ,SP,FE,OT,  
OT,OT,EJ,OT,OT,EJ,OT,OT,OT,EJ  
]

emotion3 = [  
OT,OT,EJ,OT,OT,EJ,OT,EJ,FE,FE,  
EJ,EJ,SP,SP,SN,OT,EJ,OT,OT,EJ,  
EJ,FE,SN,EJ,OT,OT,EJ,EJ,FE,EJ,  
SN,EJ,EJ,OT,EJ,FE,SN,OT,EJ,OT,  
SN,EJ,EJ,SN,EJ,EJ,SN,OT,EJ,EJ,  
OT,FE,SN,AG,OT,OT,EJ,OT,EJ,OT,  
OT,SP,SP,OT,SN,EJ,EJ,SN,OT,OT,  
SP,SP,EJ,OT,OT,EJ,OT,OT,DG,SN,  
OT,AG,EJ,EJ,OT,SN,SP,FE,SN,EJ,  
EJ,EJ,EJ,OT,OT,SP,FE,SP,EJ,FE  
]

Sau khi gán nhãn xong, tạo một dataframe gồm 2 cột “Emotion” và “Sentence” lần lượt chứa thông tin về nhãn cảm xúc và câu bình luận. Viết vào file csv bằng to\_csv của pandas, mã hóa bằng “utf-8-sig” và chứa trong folder “dataList” vừa tạo.

```
dataList1 = {"Emotion": emotion1, "Sentence": data1}
df1 = pd.DataFrame(dataList1)
df1.to_csv(storageDir + "dataList1.csv", encoding = "utf-8-sig")
df1.head()
```

0	Other	Số người mún ah Khoa cho xem mặt cameraman dựa...
1	Enjoyment	Kết mỗi câu : “Hầu như mọi người trên thế giới...
2	Enjoyment	9:30 giống trẻ trâu ai cập
3	Other	Khi Khoa đi du lịch cái mà Khoa đem theo nhiều...
4	Other	Bay qua Dubai reveiw đi anh Khoa ơi.\nAi thấy ...

```
dataList2 = {"Emotion": emotion2, "Sentence": data2}
df2 = pd.DataFrame(dataList2)
df2.to_csv(storageDir + "dataList2.csv", encoding = "utf-8-sig")
df2.head()
```

0	Other	Review về Dubai đi anh Khoa. Ai đồng ý xin 1 l...
1	Enjoyment	Nhìn ông review đồ ăn mà mình chỉ biết ngậm ng...
2	Sadness	T mắc cười khúc đầu bếp chào xong cái ông Khoa...
3	Enjoyment	Anh làm quay phim 1 ngày cho caramen review đi...
4	Surprise	đi nhật bản đi ai ý giống mình xin like

```
dataList3 = {"Emotion": emotion3, "Sentence": data3}
df3 = pd.DataFrame(dataList3)
df3.to_csv(storageDir + "dataList3.csv", encoding = "utf-8-sig")
df3.head()
```

0	Other	ở ẩn độ có người nhiễm cúm corona rồi đó, ko t...
1	Other	Năm 2014 tôi được tổng cty cho đi chuyến tập h...
2	Enjoyment	Những video như thế này làm ta thêm biết ơn nh...
3	Other	Thật cảm phục bạn Khoa luôn, mạo hiểm và tính ...
4	Other	Phải nói ở ẩn độ. Người giàu thì giàu quá....c...

Khi đọc dữ liệu chỉ cần dùng `read_csv` của pandas, đồng thời mã hóa bằng “utf8”.

```
dataList1 = pd.read_csv(storageDir + "dataList1.csv", encoding = "utf8")
dataList2 = pd.read_csv(storageDir + "dataList2.csv", encoding = "utf8")
dataList3 = pd.read_csv(storageDir + "dataList3.csv", encoding = "utf8")
```

Loại bỏ cột thuộc tính dư thừa

```
dataList1.pop("Unnamed: 0")
dataList2.pop("Unnamed: 0")
dataList3.pop("Unnamed: 0")
```

### 3.2.3 Một số đóng góp (Contributions)

Một số đóng góp khác cũng được trình bày trong bài báo cáo này gồm: Tính độ dài các câu bình luận, thống kê tần suất xuất hiện, lập biểu đồ cột và biểu đồ tròn để trực quan hóa hơn về dữ liệu thu thập được từ 3 video trên.

```

lengthDataList1 = []
lengthDataList2 = []
lengthDataList3 = []
for i in range(100):
    lengthDataList1.append(len(dataList1.Sentence[i]))
    lengthDataList2.append(len(dataList2.Sentence[i]))
    lengthDataList3.append(len(dataList3.Sentence[i]))
dataList1["Length"] = lengthDataList1
dataList2["Length"] = lengthDataList2
dataList3["Length"] = lengthDataList3
def Show_HistGraph_And_PieChart(dataList):
    E_DG = dataList.loc[dataList.Emotion == DG, 'Length']
    E_EJ = dataList.loc[dataList.Emotion == EJ, 'Length']
    E_AG = dataList.loc[dataList.Emotion == AG, 'Length']
    E_SP = dataList.loc[dataList.Emotion == SP, 'Length']
    E_SN = dataList.loc[dataList.Emotion == SN, 'Length']
    E_FE = dataList.loc[dataList.Emotion == FE, 'Length']
    E_OT = dataList.loc[dataList.Emotion == OT, 'Length']
    config = dict(alpha=0.4, bins=30, edgecolor = "black")
    plt.figure(figsize=(6,3))
    plt.hist(E_DG, **config, color = 'b', label = 'Disgust')
    plt.hist(E_EJ, **config, color = 'r', label = 'Enjoyment')
    plt.hist(E_AG, **config, color = 'g', label = 'Anger')
    plt.hist(E_SP, **config, color = 'c', label = 'Surprise')
    plt.hist(E_SN, **config, color = 'm', label = 'Sadness')
    plt.hist(E_FE, **config, color = 'y', label = 'Fear')
    plt.hist(E_OT, **config, color = 'k', label = 'Other')

```

```

plt.ylabel('Frequency')
plt.xlabel('Sentence Length')
plt.legend();
Pie=[len(E_DG),len(E_EJ),len(E_AG),len(E_SP),len(E_SN),len(E_FE),
len(E_OT)]
f, Ax = plt.subplots()
Ax.pie(Pie,labels=['DG','EJ','AG','SP','SN','FE','OT'],
autopct='%1.0f%%', startangle=90)
Ax.axis('equal')

```

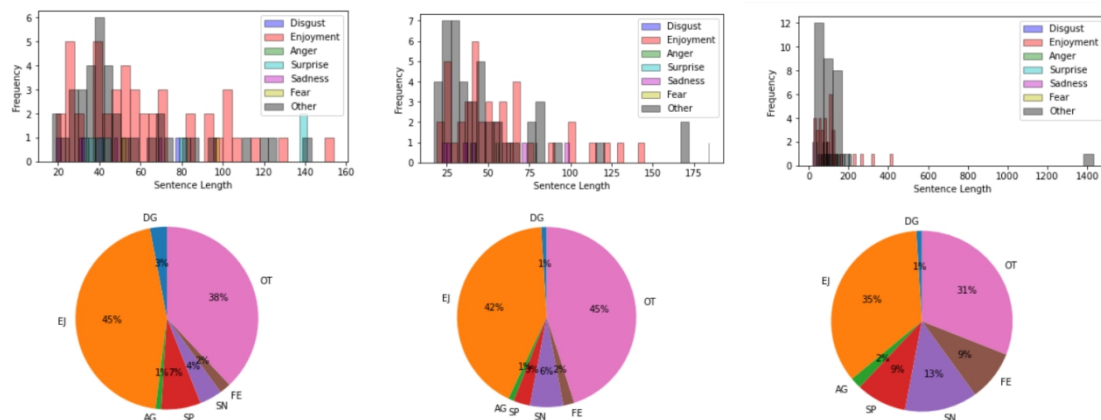
Vẽ biểu đồ với tham số đầu vào là các tập dữ liệu:

*Show\_HistGraph\_And\_PieChart(dataList1)*

*Show\_HistGraph\_And\_PieChart(dataList2)*

*Show\_HistGraph\_And\_PieChart(dataList3)*

**Kết quả:**



**Hình 2. Các biểu đồ của 3 tập dữ liệu thu được từ video kênh youtube Khoa Pug**



### 3.3 Nhận biết cảm xúc đối với văn bản trên mạng xã hội Việt Nam

#### 3.3.1 Huấn luyện mô hình từ tập dữ liệu tập huấn và thẩm định của UIT-VSMEC

Trước tiên, cần đọc các tập dữ liệu từ folder UIT-VSMEC. Mã hóa bằng “latin” sau đó gán cho các biến tương ứng.

```
dataTrainUIT = open_workbook("UIT-VSMEC/train_nor_811.xlsx", encoding_  
override='latin')  
dataTrainUIT = pd.read_excel(dataTrainUIT)  
dataValidUIT = open_workbook("UIT-VSMEC/valid_nor_811.xlsx", encoding_  
override='latin')  
dataValidUIT = pd.read_excel(dataValidUIT)  
data_TestUIT = open_workbook("UIT-VSMEC/test_nor_811.xlsx", encoding_  
override='latin')  
data_TestUIT = pd.read_excel(data_TestUIT)
```

Khởi tạo mô hình cây quyết định phân loại:

```
model = DecisionTreeClassifier()
```

Gán các biến nhãn cảm xúc Y:

```
trainY = dataTrainUIT.Emotion  
validY = dataValidUIT.Emotion  
testY = data_TestUIT.Emotion
```

Tạo hàm tiền xử lý dữ liệu để loại bỏ các từ không mang nhiều ý nghĩa, các từ ngữ thô tục, phân đoạn từ tiếng Việt, học từ vựng và tính toán idf.

```

def text_process(datasets):
    StopWords = [
        'chs','cerrrr','aaaaa','aaaaaaaa','aamir','abcxyz','ac','18','200','500','dek','t
        hg','đg','đs','đm','đuma','vl','vcl','kkk','dcm','cu','ừm','đĩ','đụ','đit','xl','lol','0
        1','10','100','11','12','13','14','15','150','17','1700','1967','20','21','22','225',
        '23','24','25','26','28','2_','2_3','30','300','3000','320','333','33333','40','40
        0','42','45','48','50','5000','580','60','63','66','75','78','80','800','81','850','9
        0','900','99','99999','_200','_5','ah','bn','c3','chg','cp','dòg','hlin','lòn','lôz','
        tđ','đkm','đkmm','đmaaaa','đmm','đmmmmm','đỹ','vcb','vclll','đụ_mẹ','trươ
        ','truen','amir','ga','1700','1967','bg','chaiii','clm','cmm','cmnl','cã','hloz','i
        mdb','kau','kbh','loz','lozzz','lozzzz','matlon','muô','nh','nhma','p30','16','25
        0','56','adm','ngươ'
    ]
    Tfidf = TfidfVectorizer(stop_words = StopWords)
    WordSeg = []
    for i in range(datasets.shape[0]):
        sentence_strip = datasets.Sentence[i].strip()
        WordSeg.append(word_tokenize(sentence_strip, format='text'))
    datasets.Sentence = WordSeg
    X = Tfidf.fit_transform(datasets.Sentence)
    print(Tfidf.get_feature_names()); print("\n\n")
    return X

```

Biến “StopWords” chứa các từ đã được cho là không mang ý nghĩa hoặc thô tục. Sau đó khởi tạo Biến “Tfidf” thực hiện véc tơ hóa TfidfVectorizer. Khởi tạo biến “WordSeg” để lưu lại các dữ liệu sau khi đã xử lý về khoảng cách bởi hàm strip() và phân đoạn từ bởi word\_tokenize của Underthesea. Với format=’text’ sẽ làm nhiệm vụ

nổi các từ có nghĩa tiếng Việt lại bằng kí tự “\_”. Tiếp theo, cập nhật lại cột “Sentence” chính là dữ liệu từ biến “WordSeg”. Cuối cùng, tiến hành cho biến “Tfidf” đã khởi tạo ban đầu, học từ vựng, phân tích số lần xuất hiện của từng thuật ngữ và chuẩn đổi về dạng ma trận tài liệu từ. Có thể sử dụng `get_feature_names()` để xem quá trình xử lý phân đoạn từ và học từ của Tfidf. Xuất ra kết quả ma trận tài liệu từ đã qua giai đoạn tiền xử lý văn bản.

Sử dụng dữ liệu tập huấn và dữ liệu xác thực từ UIT-VSMEC để tập huấn cho mô hình nhận diện cảm xúc thông qua cây quyết định phân loại đã khởi tạo ở trên học từ:

```
trainY = pd.Series(trainY, name="Emotion").to_frame()
validY = pd.Series(validY, name="Emotion").to_frame()
trainValidX = pd.concat([dataTrainUIT,
dataValidUIT]).reset_index(drop=True)
trainValidX = text_process(trainValidX)
trainValidY = pd.concat([trainY, validY]).reset_index(drop=True)
trainX, testX, trainY, testY = train_test_split(trainValidX, trainValidY,
test_size=0.33, random_state=42)
model = model.fit(trainX, trainY)
predY = model.predict(testX)
```

### Kết quả:

'anime', 'anw', 'app', 'au', 'aumobile', 'auto', 'ava', 'avatar', 'avenger', 'aw', 'awwww', 'axit', 'axit', 'a  
y', 'azz', 'ba', 'ba\_má', 'ba\_mẹ', 'babe', 'bai', 'balo', 'ban', 'ban\_tặng', 'banh', 'banking', 'bankon', 'bao',  
'bao\_bien', 'bao\_che', 'bao\_cấp', 'bao\_dai', 'bao\_dung', 'bao\_giờ', 'bao\_lâu', 'bao\_nhiều', 'bao\_tay', 'bao\_xa',  
'bar', 'bat', 'baton', 'bay', 'bb', 'bcs', 'be', 'bede', 'beep', 'ben', 'beng', 'best', 'bh', 'bi', 'bi\_đất', 'b  
ia', 'bieber', 'biet', 'bim', 'binh', 'bit', 'biê', 'biên', 'biên\_chê', 'biên', 'biên\_dạng', 'biên\_thái', 'biên  
động', 'biết', 'biết\_bao', 'biết\_bao\_nhiều', 'biết\_chừng\_nào', 'biết\_mây', 'biết\_sao', 'biết\_đầu', 'biết\_ơn', 'b  
iên', 'biểu\_cảm', 'biện\_pháp', 'biệt\_tích', 'bjt', 'block', 'blood', 'bo', 'bome', 'bon', 'bon\_chen', 'bong', 'b  
ong\_bóng', 'bonus', 'boom', 'boss', 'boxing', 'bro', 'bt', 'bth', 'bthg', 'buff', 'bung', 'buôn', 'buông', 'buồ  
i', 'buồn', 'buồn\_buôn', 'buồn\_bực', 'buồn\_cười', 'buồn\_ngủ', 'buồn\_nôn', 'buồn\_tê', 'buông\_trứng', 'buổi', 'b  
à', 'bà\_con', 'bài', 'bài\_học', 'bài\_tập', 'bàn', 'bàn\_ghế', 'bàn\_luận', 'bàn\_tay', 'bàn\_tay\_trắng', 'bàn\_thờ',  
'bào', 'bày', 'bày\_đặt', 'bá', 'bá\_đạo', 'bác', 'bác\_học', 'bác\_sĩ', 'bái\_phục', 'bám', 'bán', 'bán\_lẻ', 'bán\_l  
ô', 'bánh', 'bánh\_bèo', 'bánh\_chung', 'bánh\_kem', 'bánh\_kẹo', 'bánh\_trung\_thu', 'bánh\_trắng', 'báo', 'báo\_chí',

**Hình 3. Một đoạn trích từ tập dữ liệu UIT-VSMEC sau quá trình tập huấn**

### 3.3.2 Số liệu hiệu suất

Từ đó, ta sẽ tính toán được các số liệu hiệu suất cho test set của UIT-VSMEC:

```
Accuracy = []
Precision = []
F1_Score = []
Accuracy.append(accuracy_score(testY, predY))
F1_Score.append(f1_score(testY, predY, average='weighted')) # Could change
to None, 'weighted', 'micro', 'macro'
Precision.append(precision_score(testY, predY, average='weighted'))
averageAccuracy = sum(Accuracy)/len(Accuracy)
print("Average of Accuracy:", averageAccuracy)
averageF1_Score = sum(F1_Score)/len(F1_Score)
print("Average of F1_Score:", averageF1_Score)
averagePrecision = sum(Precision)/len(Precision)
print("Average of Precision:", averagePrecision)
```

**Kết quả:**

```
Average of Accuracy: 0.358600583090379
Average of F1_Score: 0.35792786624529066
Average of Precision: 0.358923368012908
```

## 3.4 Áp dụng mô hình đã tập huấn từ tập dữ liệu UIT-VSMEC

### 3.4.1 Áp dụng mô hình đã tập huấn cho 3 tập dữ liệu từ video youtube

Với mô hình đã được tập huấn từ tập dữ liệu UIT-VSMEC, ta sẽ lại tiếp tục áp dụng nó để tập huấn cho 3 tập dữ liệu đã thu được từ Youtube.

```
dataListY1 = dataList1.Emotion
dataListY2 = dataList2.Emotion
dataListY3 = dataList3.Emotion
```

```

dataListX1 = text_process(dataList1)
trainX1, testX1, trainY1, testY1 = train_test_split(dataListX1, dataListY1,
test_size=0.33, random_state=42)
model = model.fit(trainX1, trainY1)
predY1 = model.predict(testX1)

```

```

dataListX2 = text_process(dataList2)
trainX2, testX2, trainY2, testY2 = train_test_split(dataListX2, dataListY2,
test_size=0.33, random_state=42)
model = model.fit(trainX2, trainY2)
predY2 = model.predict(testX2)

```

```

dataListX3 = text_process(dataList3)
trainX3, testX3, trainY3, testY3 = train_test_split(dataListX3, dataListY3,
test_size=0.33, random_state=42)
model = model.fit(trainX3, trainY3)
predY3 = model.predict(testX3)

```

### **Kết quả:**

an', 'cameramen', 'canada', 'cao', 'cao sô', 'caramel', 'caramen', 'chiến tranh', 'cho', 'chuyên', 'chuân', 'chà o hỏi', 'chào khoa', 'chân', 'chém', 'chén', 'chú', 'chúc', 'chơi', 'chua', 'chạy', 'chảy', 'chăm', 'chât', 'chă c', 'chỉ', 'chị', 'chịu', 'chốt', 'chủ', 'chủ tịch', 'chữ', 'clip', 'clips', 'coi', 'con', 'cute', 'càng', 'càng ngày càng', 'cá', 'cá', 'cái', 'cân nhắc', 'câu', 'còn', 'có', 'có thể', 'cò', 'công phượng', 'cùng', 'cũ', 'c ũng', 'cơ mà', 'com', 'cung', 'cười', 'cười cười', 'cường lực', 'cạnh', 'cà', 'cảm giác', 'cảm thấy', 'cảnh sá t', 'cấp', 'cầm', 'cần', 'cập', 'cậu', 'có', 'của', 'củng', 'cứ', 'cứng', 'dang', 'dao', 'do', 'du', 'du lịch', 'dubai', 'danh', 'dân tộc', 'dê', 'dê sô', 'dê thương', 'dịch', 'dốt', 'dữ', 'dự', 'em', 'english', 'fan', 'gh e', 'ghen', 'ghê', 'ghe', 'giao thông', 'giao tiếp', 'giàu', 'giúp', 'giơ', 'giả vờ', 'giọng', 'giống', 'giờ', 'gái', 'gê', 'gi', 'gân', 'gặp', 'haha', 'hahaa', 'hay', 'hc', 'hi', 'hihi', 'hiêm', 'hiếu khách', 'hiền', 'hiê u', 'hiện tại', 'hok', 'how', 'hài', 'hài hước', 'hình', 'hình như', 'hóa', 'hót', 'hò', 'hôm', 'hông', 'hơn',

**Hình 4. Một đoạn trích từ tập dữ liệu của Video Youtube 1 sau quá trình tập huấn**

'bo', 'bo\_kobe', 'bun', 'buon', 'ba', 'ban', 'bai\_say', 'bem', 'ben', 'binh\_dan', 'bi\_may', 'bit', 'bo', 'bu', 'ban', 'ban', 'bat', 'bep', 'bi', 'bich', 'bon', 'bo', 'bua', 'ca', 'cam', 'cameraman', 'cameramen', 'caramen', 'cha', 'chi', 'chia', 'cho', 'chu\_dao', 'chuan', 'chao', 'chinh', 'chu', 'chuc', 'choi', 'chua', 'chay', 'cham', 'chat', 'chac', 'chan', 'chet\_cui', 'chi', 'chi\_thien', 'chi', 'chu\_tich', 'chu', 'chung', 'chu', 'ciet', 'cli', 'p', 'cmt', 'coi', 'coke', 'corona', 'cu\_dor', 'cube', 'cung\_cach', 'cuoi\_cung', 'cuoc\_doi', 'cang', 'ca', 'cac', 'cach', 'cai', 'cai\_rup', 'cau', 'cay', 'con', 'co', 'co\_may', 'co', 'cung', 'com', 'cui', 'cui\_sac', 'cai', 'cam\_nghi', 'cam\_nhan', 'cam\_xuc', 'cat', 'can', 'can\_canh', 'can', 'co', 'cua', 'cu', 'cu\_chi', 'da', 'dc', 'de', 'cor', 'di', 'du\_lich', 'dubai', 'dy', 'dai', 'dam', 'dong', 'duong', 'dang', 'dai', 'dich', 'em', 'fan', 'food', 'ghien', 'ghet', 'ghet', 'gia\_vo', 'giay', 'giong', 'giới\_thiệu', 'gái', 'gi', 'gat', 'gan', 'gan\_gui', 'gam', 'ga

**Hình 5. Một đoạn trích từ tập dữ liệu của Video Youtube 2 sau quá trình tập huấn**

hole', 'bug', 'bac', 'banh', 'bao\_chi', 'bay', 'bay\_gioi', 'bay\_h', 'ben', 'binh\_an', 'binh\_yen', 'bip', 'bit', 'bung', 'bung\_phat', 'ban', 'bao', 'bao\_trong', 'bang', 'benh', 'bi', 'bich', 'bo\_xa', 'bo\_me', 'buc\_xuc', 'ca', 'c', 'cam', 'camera', 'cameraman', 'can\_dam', 'cao\_sang', 'caramen', 'ch', 'chet', 'chi', 'chi\_chit', 'cho', 'chu', 'yen', 'chan', 'chan\_that', 'chin', 'chong\_may', 'chu', 'chuc', 'choi', 'chanh', 'chao', 'chac', 'chang', 'chet', 'chi', 'chi', 'chui', 'cho', 'choi', 'choi', 'chui', 'chuc', 'chu\_de', 'chu', 'clip', 'cmt', 'co', 'coi', 'con', 'con\_ngu', 'oi', 'corona', 'cov', 'covit', 'cuoc\_song', 'ca\_nhan', 'cac', 'cach', 'cai', 'cam\_on', 'cam\_on\_khoa', 'cui', 'con', 'co', 'co\_le', 'co\_le\_ahn\_dor', 'co\_thet', 'co\_vo', 'co', 'cong\_viec', 'cum\_corona', 'cang', 'cung', 'ca', 'cam\_thay', 'cam\_tay', 'cam\_on', 'canh', 'cam', 'cau', 'cau\_mong', 'canh\_than', 'cap\_co\_dai', 'co', 'co', 'c

**Hình 6. Một đoạn trích từ tập dữ liệu của Video Youtube 3 sau quá trình tập huấn**

### 3.4.2 Số liệu hiệu suất

Sau khi áp dụng mô hình vào tập huấn cho 3 video Youtube, ta sẽ tính toán được số liệu hiệu suất của chúng như sau:

```
Accuracy1 = []
Precision1 = []
F1_Score1 = []
Accuracy1.append(accuracy_score(testY1, predY1))
F1_Score1.append(f1_score(testY1, predY1, average='micro')) # Could change
to None, 'weighted', 'micro', 'macro'
Precision1.append(precision_score(testY1, predY1, average='micro'))
averageAccuracy1 = sum(Accuracy1)/len(Accuracy1)
print("Average of Accuracy 1:", averageAccuracy1)
averageF1_Score1 = sum(F1_Score1)/len(F1_Score1)
print("Average of F1_Score 1:", averageF1_Score1)
averagePrecision1 = sum(Precision1)/len(Precision1)
print("Average of Precision 1:", averagePrecision1)
```

```
Accuracy2 = []
Precision2 = []
F1_Score2 = []
Accuracy2.append(accuracy_score(testY2, predY2))
F1_Score2.append(f1_score(testY2, predY2, average='micro')) # Could change
to None, 'weighted', 'micro', 'macro'
Precision2.append(precision_score(testY2, predY2, average='micro'))
averageAccuracy2 = sum(Accuracy2)/len(Accuracy2)
print("Average of Accuracy 2:", averageAccuracy2)
averageF1_Score2 = sum(F1_Score2)/len(F1_Score2)
print("Average of F1_Score 2:", averageF1_Score2)
averagePrecision2 = sum(Precision2)/len(Precision2)
print("Average of Precision 2:", averagePrecision2)
```

```
Accuracy3 = []
Precision3 = []
F1_Score3 = []
Accuracy3.append(accuracy_score(testY3, predY3))
F1_Score3.append(f1_score(testY3, predY3, average='micro')) # Could change
to None, 'weighted', 'micro', 'macro'
Precision3.append(precision_score(testY3, predY3, average='micro'))
averageAccuracy3 = sum(Accuracy3)/len(Accuracy3)
print("Average of Accuracy 3:", averageAccuracy2)
averageF1_Score3 = sum(F1_Score3)/len(F1_Score3)
print("Average of F1_Score 3:", averageF1_Score3)
averagePrecision3 = sum(Precision3)/len(Precision3)
print("Average of Precision 3:", averagePrecision3)
```

**Kết quả:**

*Average of Accuracy 1: 0.36363636363636365*

*Average of F1\_Score 1: 0.36363636363636365*

*Average of Precision 1: 0.36363636363636365*

*Average of Accuracy 2: 0.5454545454545454*

*Average of F1\_Score 2: 0.5454545454545454*

*Average of Precision 2: 0.5454545454545454*

*Average of Accuracy 3: 0.5454545454545454*

*Average of F1\_Score 3: 0.21212121212121215*

*Average of Precision 3: 0.21212121212121213*



## CHƯƠNG IV: TỔNG KẾT

Bài báo cáo này đã giải thích chi tiết từng bước triển khai từ thu thập dữ liệu từ video trên mạng xã hội Youtube, gán nhãn cảm xúc cho từng bình luận, tập huấn và thống kê các số liệu hiệu suất. Với việc sử dụng một số thư viện như Selenium trong việc thu thập dữ liệu, các kỹ thuật thông dụng trong việc trích xuất và xử lý dữ liệu của văn bản của thư viện Scikit-learn, Underthesea. Các kết quả thu được đã giúp cho việc tìm hiểu về cách thức khai phá dữ liệu văn bản. Tuy việc gán nhãn ở đây vẫn là thủ công, nhưng các khái niệm và phương pháp đã được tìm hiểu kĩ càng nhất có thể. Bên cạnh đó, độ chính xác mà nhóm đã cố gắng thực hiện hết sức nhưng vẫn chưa được thực sự tốt.

Qua đây, nhóm đã hiểu khái quát hơn về các công nghệ mới hiện nay cũng như những ứng dụng thực tiễn của việc trích xuất đối tượng văn bản, xử lý văn bản trong nhận diện cảm xúc. Trong tương lai, nhóm có thể sẽ tìm hiểu thêm một số phương pháp, kỹ thuật khác trong việc phân tích dữ liệu cảm xúc của những bình luận từ nhiều nguồn hơn, tổng hợp dữ liệu và dự đoán, đánh giá được sở thích của người dùng khi xem các video trên mạng xã hội hay đánh giá về mức độ, trình độ văn hóa của họ.

# TÀI LIỆU THAM KHẢO

## **Tài liệu Sách và Báo:**

[1]: J.Han, M.Kamber, J.Pei, [2011], Data Mining Concepts and Techniques 3<sup>rd</sup> Edition, Illinois University, Urbana-Champaign, 83-123.

[2]: S.A.Alasadi, W.S.Bhaya , [2017], Review of data preprocessing techniques in data mining, College of Information Technology, Iraq, 4102-4107.

[3]: Scikit-learn Developers, [2020], Scikit-learn User guide - Release 0.23.2, 1930-1949.

[4] V.A.Ho, D.H.-C.Nguyen, D.H.Nguyen, L.T.-V.Pharm, D.V.Nguyen, K.V.Nguyen, N.L-T.Nguyen, [2019], Emotion Recognition for Vietnamese Social Media Text.

## **Tài liệu Internet:**

[5]: <https://www.geeksforgeeks.org/feature-extraction-techniques-nlp/>

[6]: <https://www.geeksforgeeks.org/data-preprocessing-in-data-mining/>

[7]: [https://scikit-learn.org/stable/modules/feature\\_extraction.html](https://scikit-learn.org/stable/modules/feature_extraction.html)

[8]: <https://programmersought.com/article/61456088337/>

[9]: <https://pypi.org/project/underthesea/>

---