# THAI TEXT EMOTION DETECTION

Machine Learning Final Project

# OUR TEAM

**64011363**

Chalermwut
Jutanit

**64011380**

Chonlaphat
Ingkharratphithakon

**64011551**

Phraechanok
Rimdisit

# TABLE OF CONTENT


01 Objective
Page 3-4

02 Process
Page 5-6

03 Library Used
Page 7-10

04 Data
Page 11-13

05 Model
Page 14-16

06 Result
Page 17-20

07 Application
Page 21-22

# 01

# OBJECTIVE

# Objectives


- Further analysis in other related field such as marketing
- For understanding emotion in thai language

# 02

# PROCESS

# PROCESS

STEP 1 — CREATE DATA

STEP 2 — IMPORT DATA

STEP 3 — EXPLORE DATA

STEP 4 — PREPROCESS DATA

STEP 5 — CREATE MODEL

# PROCESS

## STEP 1

### CREATE DATA

- 0 = SAD
- 1 = JOY AND HAPPINESS
- 2 = LOVE
- 3 = ANGER
- 4 = FEAR
- 5 = SURPRISE

- We create the data by gathering data from x.com

- Ask an opinion on our friends in each sentence

# PROCESS

## IMPORT DATA

STEP 2

## IMPORT

- Save excel as csv.
- Export the saved file to Github.
- Use google collab to import the saved data from Github.

# PROCESS

## STEP 3

## EXPLORE DATA

## EXPLORE

- Check for error and missing values.
- Check data size.

# PROCESS

## PREPROCESS DATA

**STEP 4**

## PREPROCESS

- Import PYTHAINLP library.
- Change emoji to thai words using stopword from THAINLP library.
- Tokenize thai words - to move all the emoji to the back and eliminating the meaningless word as well as split data into words.

# PROCESS

STEP 5

## CREATE MODEL

## MODEL

- Use countvectorizer to create bag of words for Multinomial NB and Logistic Model.
- Use TF-IDF for Logistic regression.
- Make Multinomial NB and Logistic Model.

03

# LIBRARY USED

# LIBRARY USED

## SKLEARN

Provides simple and efficient tools for data analysis and modeling.

## PYTHAINLP

Library used for tokenizing thai language.

## PANDAS

Library used for creating DataFrames for training set.

# LIBRARY USED

## MATPLOTLIB

Helps generate plots and charts for the model.

## YELLOWBRICK

Library built on top of Matplotlib design to enhance ML model

## NUMPY

Library used for numerical computing in a large multi-dimensional arrays

# LIBRARY USED

## STREAMLIT

Allows developers to turn data scripts into shareable web apps with minimal code.

## JOBLIB

Used to save model to cloud

# 04

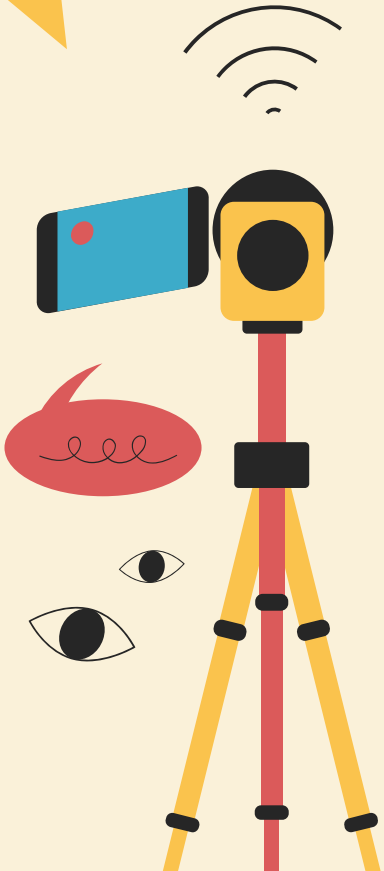
# DATA

DATA from X.com

# DATA

## DATA DESCRIPTION

- 0 = SAD
- 1 = JOY AND HAPPINESS
- 2 = LOVE
- 3 = ANGER
- 4 = FEAR
- 5 = SURPRISE



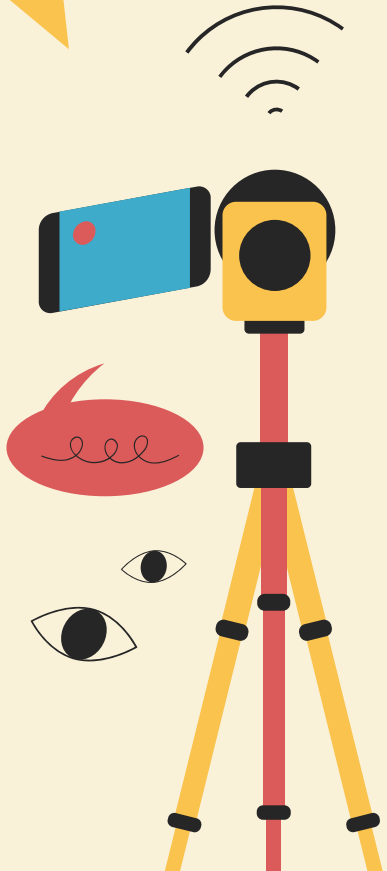
# DATA

## CRITERIA

- The data we are choosing is a Data in X.com which is the most frequent community in Thai needs and young adults to comment their thought on which has a unique slag and language
- Hashtag - we select hashtag according to our target such as #เศร้า #รัก and a trendy hashtag such as #อุงเอิง #โยเกิร์ต #พีเค
- The period of gathering is a recent 2-3 weeks (25 Feb-6 Mar) (for an update data - as a new slay is always created)
- There are 315 rows of data with 6 columns
- We manually label the opinion on each sentence emotion expression by asking our friend to comment o the opinion.

# DATA from X.com

| | Text | ans1 | ans2 | ans3 | ans4 | ans5 | Final | Clean_Text |
|---|---|---|---|---|---|---|---|---|
| 0 | ครูอึ้ง! โดนเซอร์ไพรส์ แต่ดันเจอเซอร์ไพรส์กลับ.. | 5 | 5 | 5 | 5 | 5 | 5 | ครูอึ้ง! โดนเซอร์ไพรส์ แต่ดันเจอเซอร์ไพรส์กลับ.. |
| 1 | วันนี้พระจันทร์น่ารักมาก :-) | 1 | 2 | 2 | 2 | 1 | 2 | วันนี้พระจันทร์น่ารักมาก :-) |
| 2 | ฮือฮา! 'พระพุทธรูปสิ่งศักดิ์สิทธิ์' ที่อยู่ในส... | 5 | 5 | 5 | 5 | 5 | 5 | ฮือฮา! 'พระพุทธรูปสิ่งศักดิ์สิทธิ์' ที่อยู่ในส... |
| 3 | "กลัวเธอจะหนีไป..กลัวเธอจะใจร้าย" | 4 | 4 | 4 | 0 | 4 | 4 | "กลัวเธอจะหนีไป..กลัวเธอจะใจร้าย" |
| 4 | มีคนแปลกหน้า เข้ามาหาเรื่องเรา | 3 | 3 | 3 | 3 | 3 | 3 | มีคนแปลกหน้า เข้ามาหาเรื่องเรา |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 310 | มันหนึบคือเหี้ยไรว่ะ รสชาสเหมือนรองเท้านักเรีย... | 3 | 3 | 3 | 0 | 3 | 3 | มันหนึบคือเหี้ยไรว่ะ รสชาสเหมือนรองเท้านักเรีย... |
| 311 | มองผ่านเหมือนแมลงสาบบินอ่ะ กัว แง | 4 | 4 | 4 | 4 | 4 | 4 | มองผ่านเหมือนแมลงสาบบินอ่ะ กัว แง |
| 312 | รู้เลยบ้านนี้ใครใหญ่สุด กัว | 4 | 4 | 4 | 4 | 4 | 4 | รู้เลยบ้านนี้ใครใหญ่สุด กัว |
| 313 | บาร์โค้ดทำถึงเกินอะ กัว😭😭😭😭 | 4 | 4 | 4 | 4 | 4 | 4 | บาร์โค้ดทำถึงเกินอะ กัว😭😭😭😭 |
| 314 | ในไอจีมาลงรูปรองเท้าผูกกันเพิ่มด้วย กัว อะ 😭😭😭... | 4 | 4 | 4 | 4 | 4 | 4 | ในไอจีมาลงรูปรองเท้าผูกกันเพิ่มด้วย กัว อะ 😭😭😭... |



# WHY THAI TEXT

Thai sentence is not straightforward

05

# MODEL

# Testing and Training dataset

```
x_train,x_test,y_train,y_test = train_test_split(X,y,test_size=0.2,shuffle=True, stratify = y,random_state=2002)
```

- We would like to training the dataset as much as possible so we set the training dataset to be 80% with shuffle and stratify the dataset

# MODELS

## Multinomial NB

There are many category output not as binomial and calculate from probability with no requirement of features

## Logistic Regression

It is classification that can find probability and can do multinomial with no requirement of feature.

# MODELS

## Multinomial NB

There are many category output not as binomial.

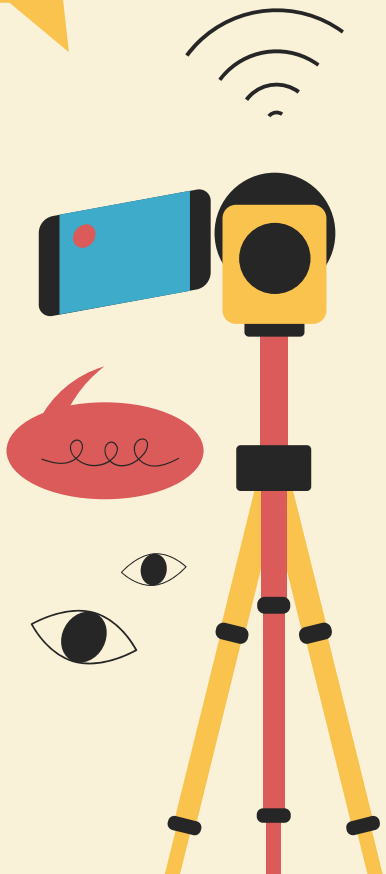
Step:

1. We tokenized the data using THAINLP library
2. Splitting data into testing and training dataset
3. Using countvectorizer to count the frequency of the words
4. Then create the model.

## Why use multinomial NB instead of other classification model?

Multinomial NB could handle the multiple class classification since there are 6 targets in our model

- 0 = SAD
- 1 = JOY AND HAPPINESS
- 2 = LOVE
- 3 = ANGER
- 4 = FEAR
- 5 = SURPRISE



# MODELS

Step:

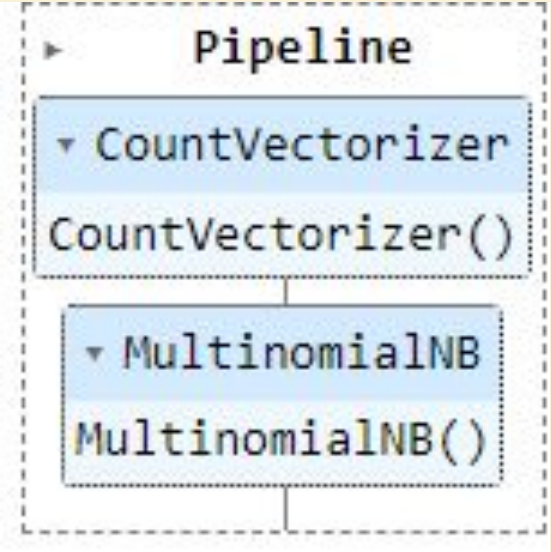
1. We tokenized the data using THAINLP library
2. Splitting data into testing and training dataset
3. Using countvectorizer to count the frequency of the words
4. Use TF-IDF to eliminate the frequent word appeared
5. Then create the model.

## Logistic Regression

It is classification that can find probability and can do multinomial.

# PIPELINE

- Automate the ML workflow
- Analyze the complexity of language without caring the coefficient
- Allow the creator to focus on the updated data only
- Reduce Model Error

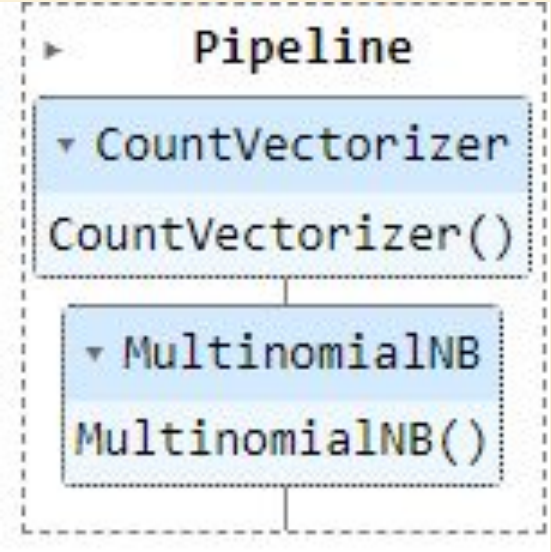


# CountVectorizer

CountVectorizer is a bag of word that we use to count the word

**Why using countvectorizer instead of other vectorizer?**

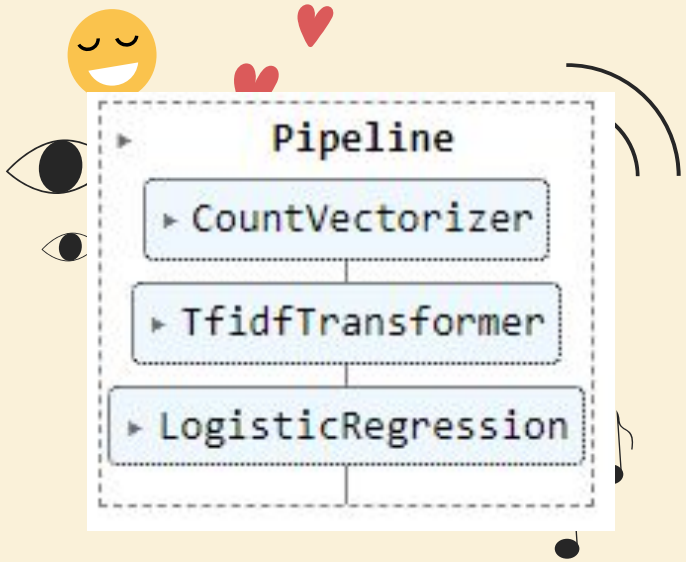
- Basic and common use vectorizer that help the ML to better understanding the model



# TF-IDF

We use TF-IDF function as

- it eliminate the frequent word appeared
- Suitable with logistic regression
- Making more understanding and more accuracy in Logistic Regression



06

# RESULT

# Multinomial NB

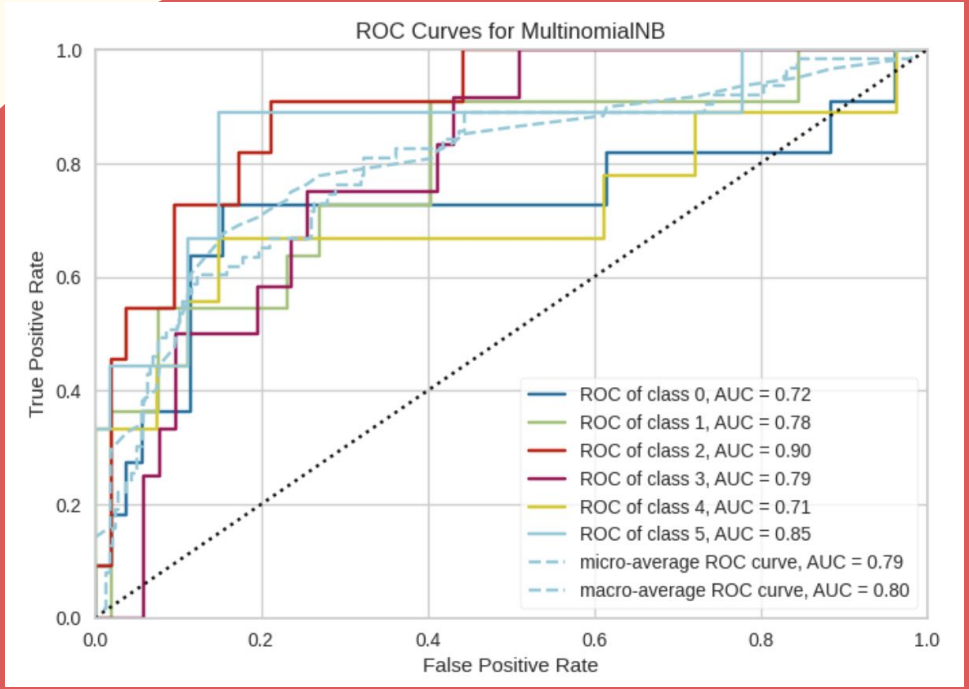
```
Classification Report:
              precision    recall  f1-score   support

         Sad       0.53      0.73      0.62        11
       Happy       0.50      0.27      0.35        11
        Love       0.64      0.64      0.64        11
       Angry       0.42      0.67      0.52        12
        Fear       0.50      0.33      0.40         9
    Surprise       0.67      0.44      0.53         9

    accuracy                           0.52        63
   macro avg       0.54      0.51      0.51        63
weighted avg       0.54      0.52      0.51        63

F1 score = 0.5090252989683729
Accuracy = 0.5238095238095238
Accuracy on train:   0.968

Accuracy on test:  0.524
```



# Logistic Regression

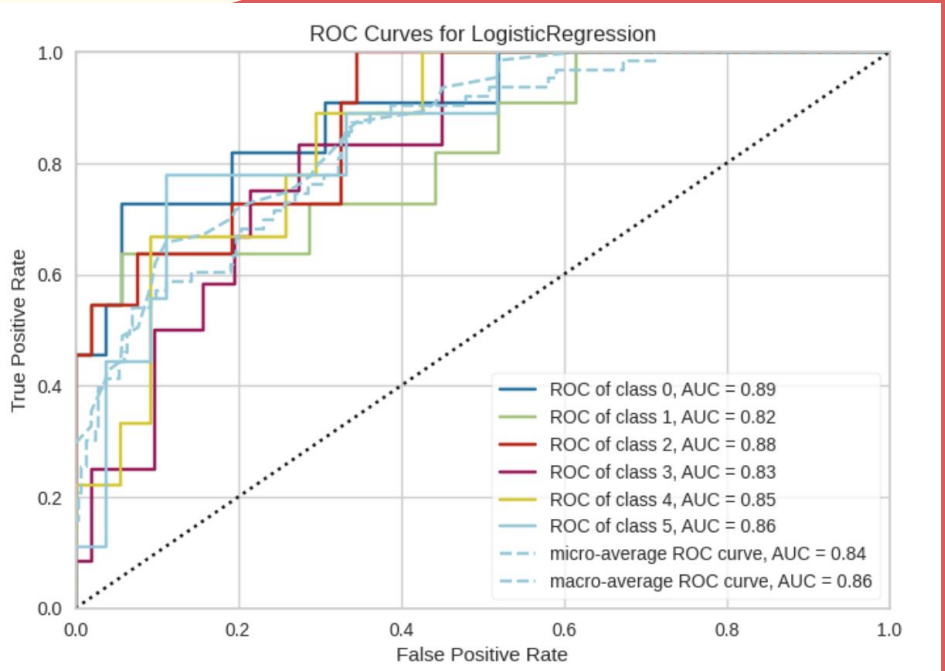
```
Classification Report:
              precision    recall  f1-score   support

         Sad       0.67      0.55      0.60        11
       Happy       0.50      0.64      0.56        11
        Love       0.50      0.64      0.56        11
       Angry       0.42      0.67      0.52        12
        Fear       0.67      0.22      0.33         9
    Surprise       1.00      0.44      0.62         9

    accuracy                           0.54        63
   macro avg       0.63      0.53      0.53        63
weighted avg       0.61      0.54      0.53        63

F1 score = 0.530807830162669
Accuracy = 0.5396825396825397
Accuracy on train:   0.992

Accuracy on test:  0.54
```



# Example Testing

```
ex1 = "แต่หนูโกดไมมหาสัยต้องให้หนูเรียนวันเสาร์ นอยๆๆๆๆ"
x = pipe_lr.predict([ex1])
emotions[x[0]]
```

```
'Happy'
```

```
ex1 = text_process(ex1)
print(ex1)
x = pipe_lr.predict([ex1])
emotions[x[0]]
```

```
แต่ หนู โก ดไม มหาสัย ต้อง ให้ หนู เรียน วัน เสาร์ นอ ย
'Angry'
```

```
y = pipe_lr.predict_proba([ex1])
y
```

```
array([[0.1823474 , 0.14842353, 0.14532417, 0.32425434, 0.10060137,
        0.09904918]])
```

```
x = pipe_nb.predict([ex1])
emotions[x[0]]
```

```
'Angry'
```

07

# APPLICATION

# Save Model

We use save the model to

- Prevent when get error and when code get disappeared
- To make an app for further analyze of our model

```
from google.colab import drive
drive.mount('/content/drive')
```

```
#Saving the model to run app
import joblib
joblib.dump(pipe_lr, open('/content/drive/MyDrive/ML Project/emotion_model.pkl','wb'))
```

# Web Application

## Thai Text Emotion Detection

### Detect Emotions In Text (Only Thai text)

Type Here

เซอไพรซ์มาก วันนี้ตอนกำลังกลับจากเขาใหญ่ เปิดเพลย์ลิสเคพ๊พเพราะแฟนเริ่มง่วงตอนขับ รันเพลง Queencard มา แฟนขึ้นเต้นได้เพราะเคยดู MV 5555555 จังหวะ take a photo ดีขึ้นท่าท่าแซะกล้อง ฮี ชูสองนิ้ว 555555 เลือกไม่ผิดคนละ

Submit

Prediction Results

Surprise

Other probability results with percentage

| | Sad | Happy | Love | Angry | Fear | Surprise |
|---|---|---|---|---|---|---|
| 0 | 13.8119 | 18.9614 | 12.4768 | 17.142 | 15.5274 | 22.0805 |

"The model shows to be 50%-55% accuracy, but still acceptable"

**CONCLUSION**

- Increase data set.
- Average the type of emotion in the data set.

**IMPROVEMENT**

# THANK YOU