AI final project

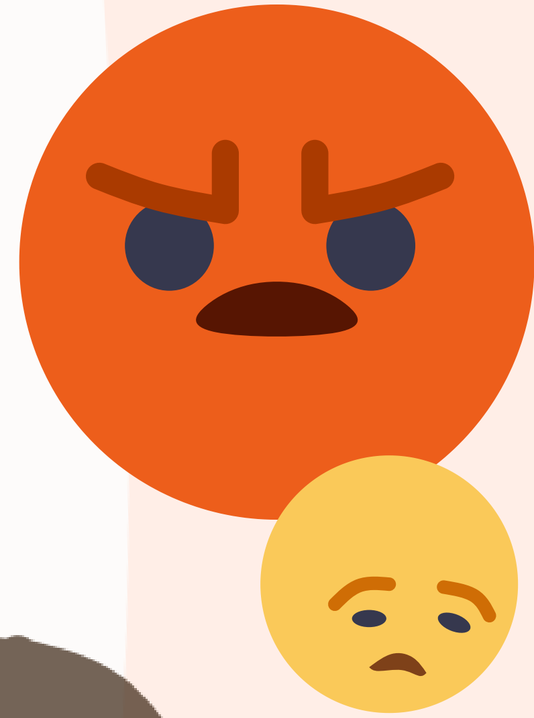
Emoji recommendation

Team 15
0813356
0813304
109550126
109550168

Introduction

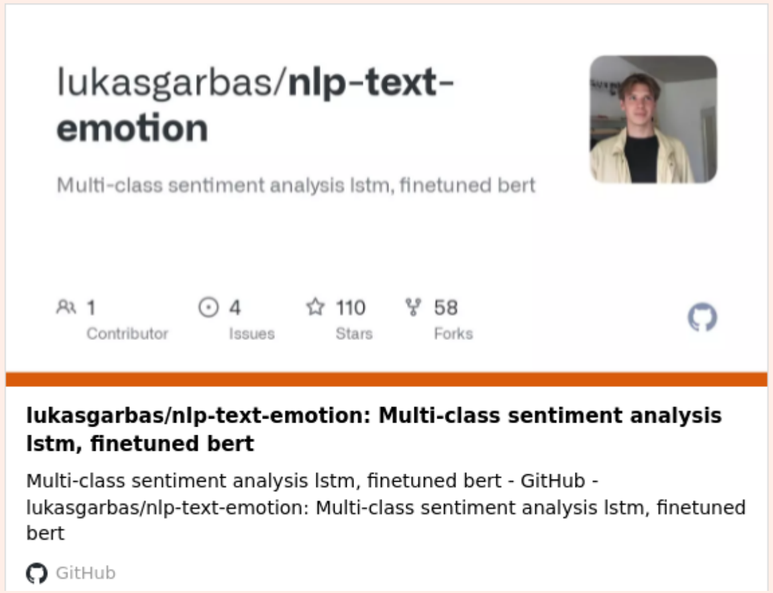
Our purpose is to train the agent to find **the best emoji choice for a sentence**, which may be applied on **communication application**.

This agent could not be very necessary, but we consider it practicable because it could help people who are not familiar with internet to choose a better emoji when they are sending messages to others.



Literature Review

Emotion Classification in Short Messages



1 **Traditional Machine Learning**

naive bayes, random forest, logistic regression, SVM

2 **Neural Network**

LSTM + w2v_wiki, biLSTM + w2v_wiki, CNN + w2v_wiki

3 **Transfer learning with BERT**

finetuned BERT

Literature Review

Approach	F1-Score
Naive Bayes	0.6702
Random Forrest	0.6372
Logistic Regression	0.6935
SVM	0.7271

Approach	F1-Score
LSTM + w2v_wiki	0.7395
biLSTM + w2v_wiki	0.7414
CNN + w2v_wiki	0.7580

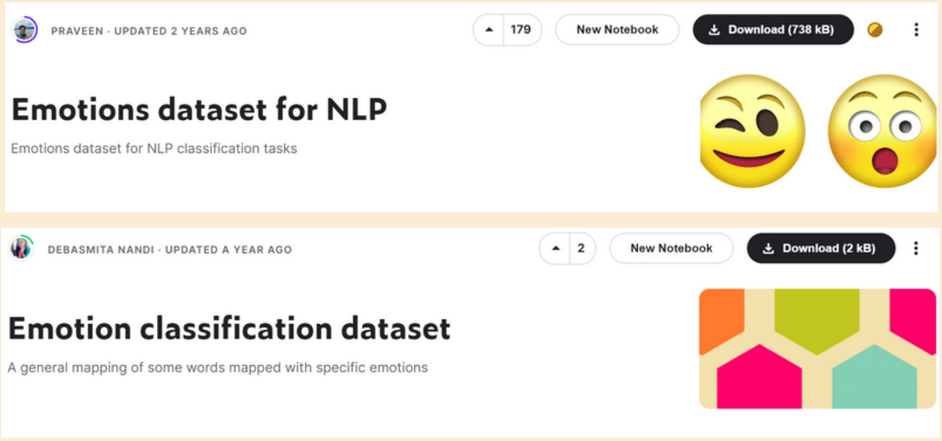
Approach	F1-Score
finetuned BERT	0.8320

So...that is why
we finally choose BERT

Dataset

Kaggle:

1. **Emotions dataset for NLP**
2. **Emotion classification dataset(Baseline)**



1 Size

#Train: 18000
#Valid: 2000

2 Distribution of classes

```
==Train==
anger: 2411
fear: 2148
joy: 6095
love: 1480
sadness: 5216
surprise: 650
```

```
==Validation==
anger: 298
fear: 225
joy: 666
love: 161
sadness: 581
surprise: 69
```

3 Preprocessing

1.remove redundant punctuation like;
,and fill into a csv having columns like "sentence"and "label"
2.mix three raw data sets, making 10% as valid and other 90% as train

Baseline

1. **rule-based method**
   use "if" expression
   if see token in our
   keyword database, then
   output corresponding emoji.

2. **random method**
   randomly pick any one emoji.

1 Keyword Database

basically, use a database from Kaggle, and we also do some modification and additional labeling.
#data: 329

feel.txt =	
sadness	147
love	61
joy	55
fear	29
anger	21
surprise	15

2 Implementation

1. count the number of keyword tokens for six emotions appearing in the sentence.
2. choose the emotion corresponding to the highest count.
(if tie, just use random.choices().)

Main Approach

In short...

- multi-class NLP model
  Classify 6 emotions：'anger', 'fear', 'joy', 'love', 'sadness', 'surprise'

- BERT：bert-base-uncased
  Levels: use the score from BERT model to classify its level
  However, it is not as powerful as we thought

reference: 李宏毅_ELMO, BERT, GPT

Main Approach

Basically, we use pre-trained models from packages.

```
from transformers import BertTokenizer, BertModel
```

attention(extract features)

↓

token[mask] prediction

↓

apply to our emotion classifier

Evaluation Metrics

- Accuracy
- Macro-F/Micro-F

$$P_{macro} = \frac{1}{n}\sum_{i=1}^{n} P_i \qquad P_{micro} = \frac{\bar{TP}}{\bar{TP} + \bar{FP}} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} TP_i + \sum_{i=1}^{n} FP_i}$$

$$R_{macro} = \frac{1}{n}\sum_{i=1}^{n} R_i \qquad R_{micro} = \frac{\bar{TP}}{\bar{TP} + \bar{FN}} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} TP_i + \sum_{i=1}^{n} FN_i}$$

$$F_{macro} = \frac{2 \times P_{macro} \times R_{macro}}{P_{macro} + R_{macro}} \qquad F_{micro} = \frac{2 \times P_{micro} \times R_{micro}}{P_{micro} + R_{micro}}$$

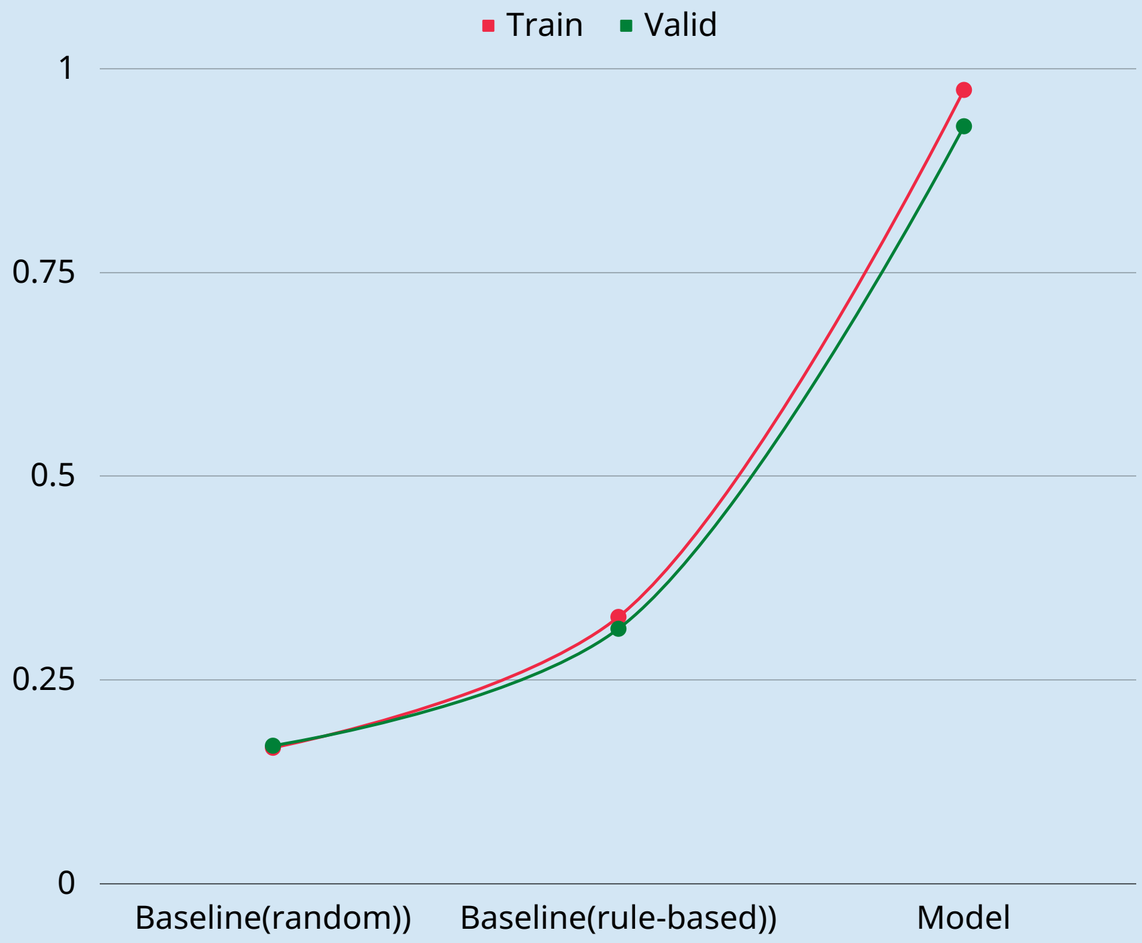
Results & Analysis

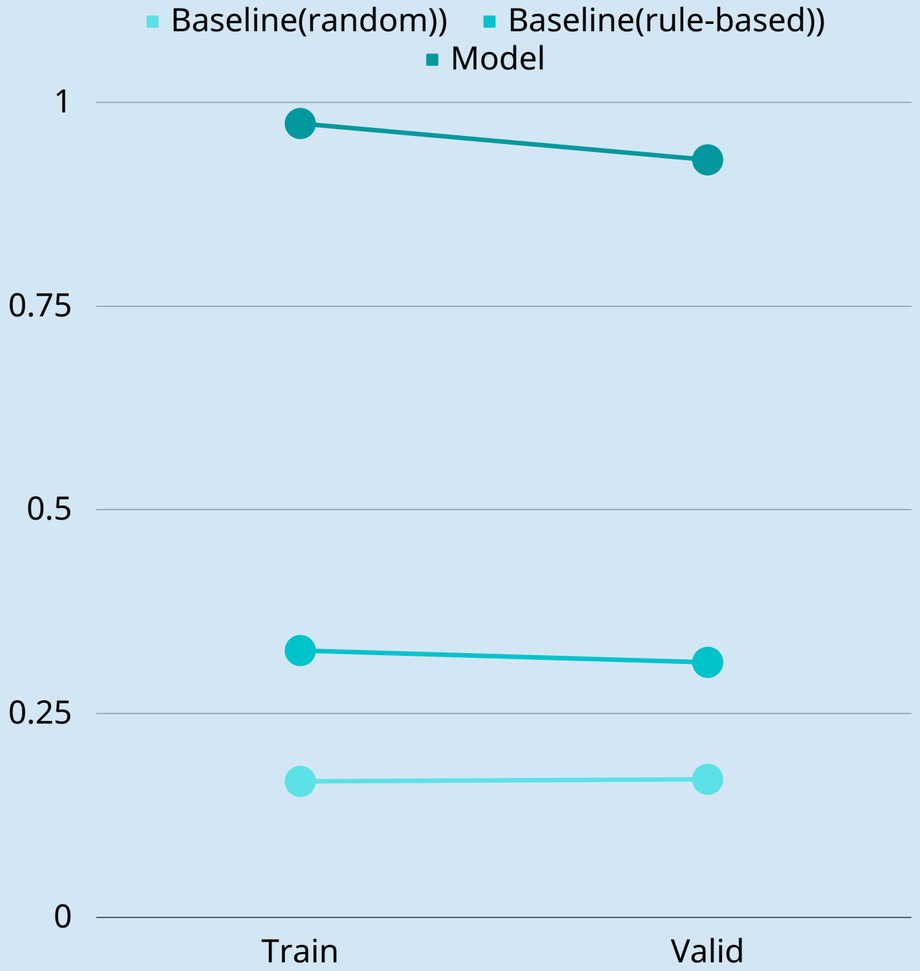
截斷至小數點後第4位

	Macro-F	Micro-F	Accuracy
Baseline (random)	**Train** (0.1673, 0.1646, 0.1659) **Valid** (0.1706, 0.1693, 0.1699)	**Train** (0.1670, 0.1670, 0.1670) **Valid** (0.1695, 0.1695, 0.1695)	**Train** 0.1670 **Valid** 0.1695
Baseline (rule-based)	**Train** (0.3095, 0.3433, 0.3256) **Valid** (0.2952, 0.3219, 0.3080)	**Train** (0.3275, 0.3275, 0.3275) **Valid** (0.3130, 0.3130, 0.3130)	**Train** 0.3275 **Valid** 0.3130
Model	**Train** (0.9567, 0.9574, 0.9571) **Valid** (0.9065, 0.8969, 0.9017)	**Train** (0.9741, 0.9741, 0.9741) **Valid** (0.9295, 0.9295, 0.9295)	**Train** 0.9741 **Valid** 0.9295

Results & Analysis

chart of Accuracy

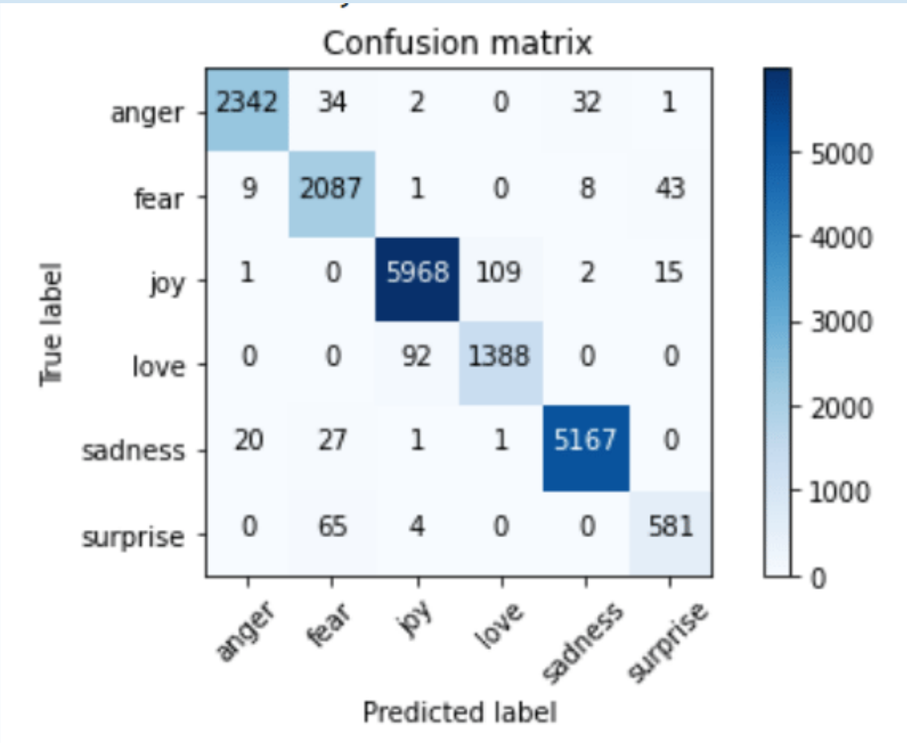




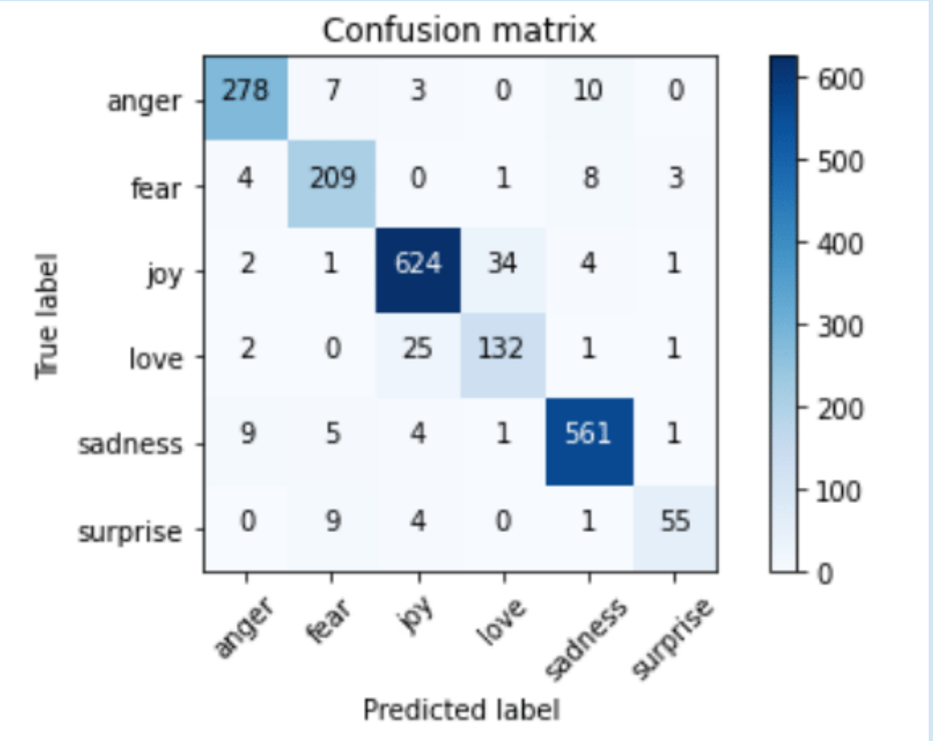
Results & Analysis

visualization of confusion matrix

Train



Valid



⚠ Error Analysis

Type	Description	Example
1	Laugh相關句子被分類為anger	I laugh out loud. → anger
2	否定句分類錯誤	I am not happy. → joy (X) I am unhappy. → sadness (O)
3	簡寫分類錯誤	LMAO → anger
4	不同對象不同情緒 無法分辨何者為自己的情緒	He is mad because I am happy. → anger
5	其他...	He has a crush on the girl. → anger Rest in peace. → joy I am on vacation. → sadness

Future Work

1. Better way to differentiate levels in specific emotion?

2. Directly use emojis as the tool for classification of database?

Code

https://github.com/yayun502/AI-final-project.git

References

Database

https://www.kaggle.com/code/praveengovi/classify-emotions-in-text-with-bert

https://www.kaggle.com/datasets/debsmitaa66/emotion-classification-dataset

Logistic Regression/n-gram/Neural Network

https://devblogs.microsoft.com/cse/2015/11/29/emotion-detection-and-recognition-from-text-using-deep-learning/

Support Vector Machines

https://arxiv.org/ftp/arxiv/papers/1708/1708.03892.pdf

https://github.com/lukasgarbas/nlp-text-emotion

Contribution of each member

Basically, for all the codes, we all work together at the same time through long hours of meetings.

As a result, we think every member has equal contribution.