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.
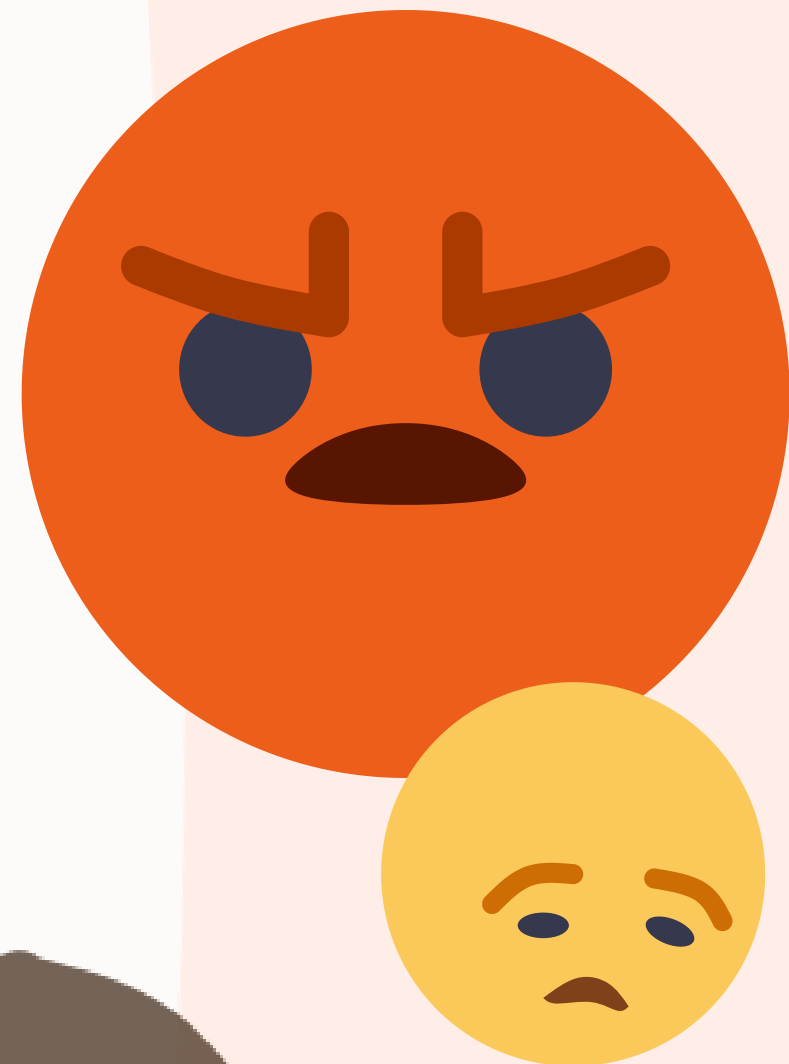
Literature Review

Emotion Classification in Short Messages
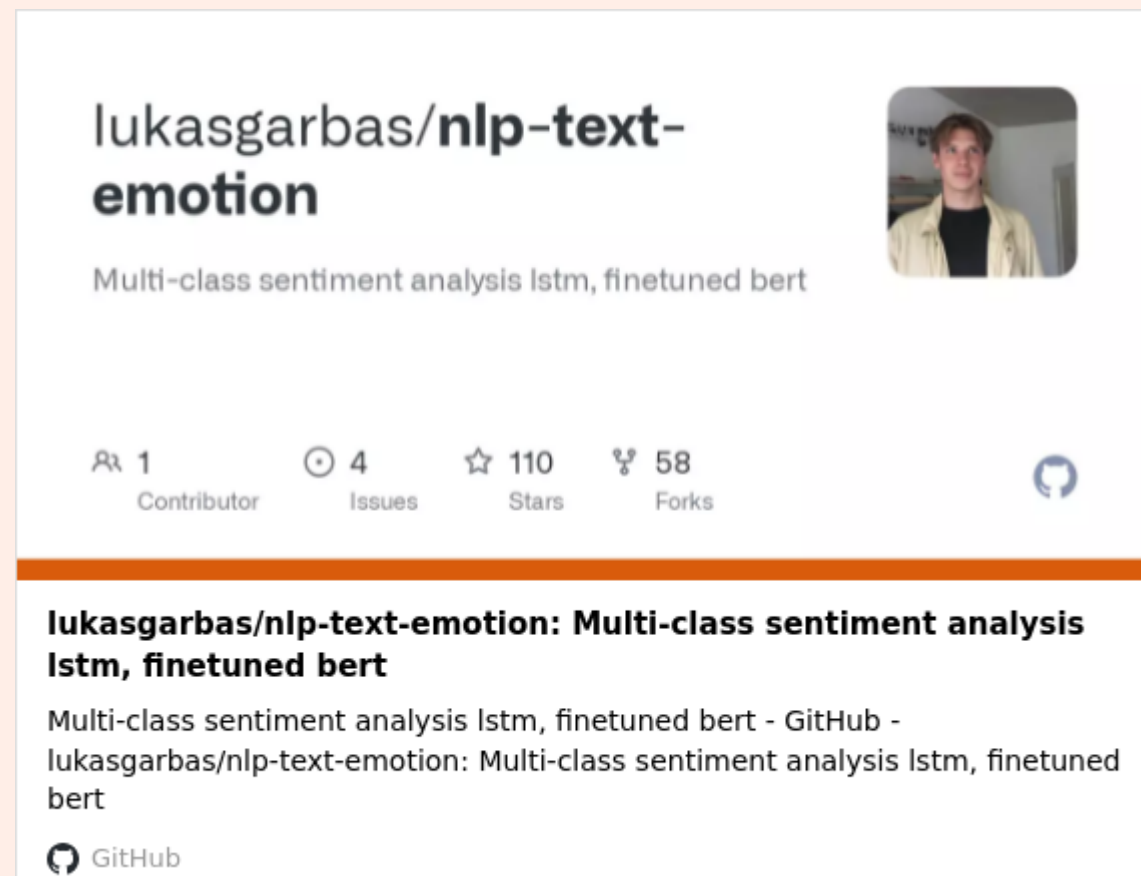
1 **Traditional Machine Learning**

naive bayes, random forest, logistic regression, SVM

2 **Neural Network**

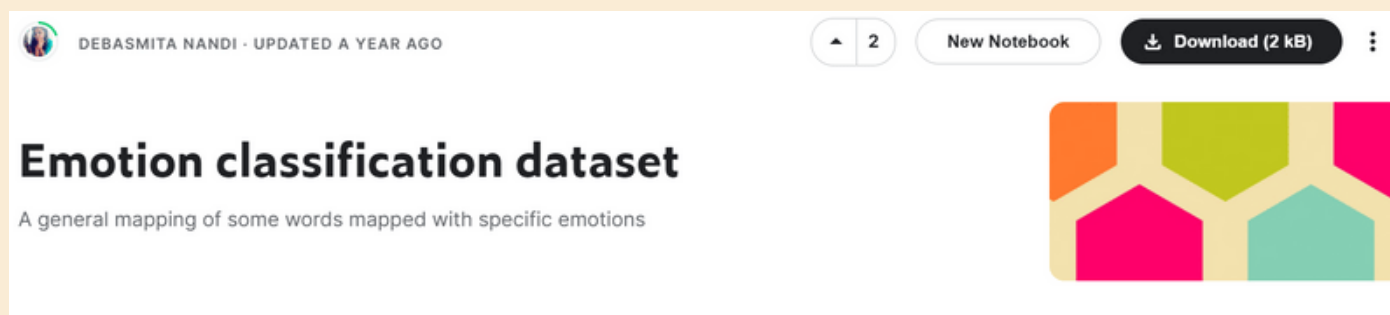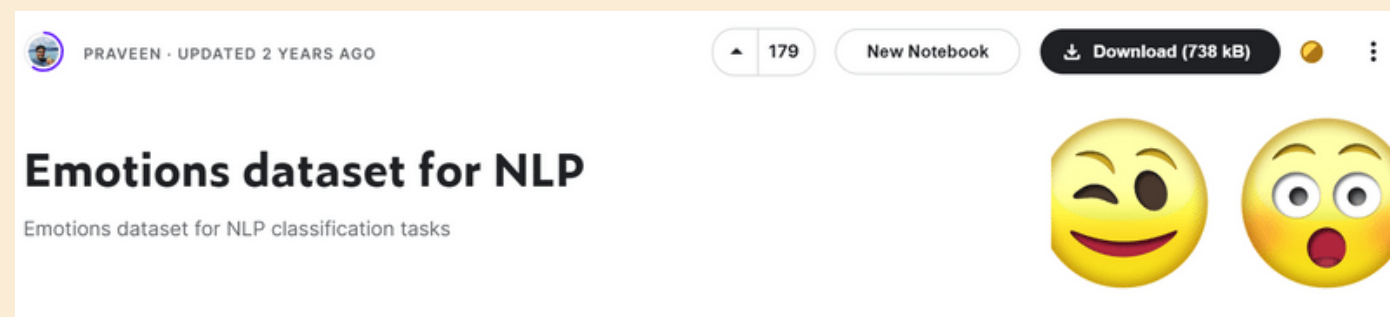LSTM + w2v_wiki, biLSTM + w2v_wiki, CNN + w2v_wiki

3 **Transfer learning with BERT**

finetuned BERT

Dataset

1 Size

#Train: 16000
#Validation: 2000
#Test: 2000

2 Distribution of classes

df_train =		df_test =		df_val =	
joy	5362	joy	695	joy	704
sadness	4666	sadness	581	sadness	550
anger	2159	anger	275	anger	275
fear	1937	fear	224	fear	212
love	1304	love	159	love	178
surprise	572	surprise	66	surprise	81

3 Preprocessing

remove redundant punctuation like;
,and fill into a csv having columns
"sentence"and "label"

Baseline

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

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

① <u>**Keyword Database**</u>

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

BERT

- 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

Evaluation Metric

- Accuracy
- Micro-F/Macro-F

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

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

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

Results & Analysis

截斷至小數點後第4位

	Micro-F			Macro-F			Accuracy		
Baseline (rule-based)	**Train** (0.3235, 0.3235, 0.3235) **Valid** (0.3225, 0.3225, 0.3225) **Test** (0.3145, 0.3145, 0.3145)			**Train** (0.3235, 0.3235, 0.3235) **Valid** (0.2940, 0.3232, 0.3079) **Test** (0.2974, 0.3268, 0.3114)			**Train** 0.3235 **Valid** 0.3225 **Test** 0.3145		
Baseline (random)	**Train** (0.1647, 0.1647, 0.1647) **Valid** (0.1700, 0.1700, 0.1700) **Test** (0.1575, 0.1575, 0.1575)			**Train** (0.1633, 0.1571, 0.1602) **Valid** (0.1557, 0.1595, 0.1576) **Test** (0.1670, 0.1532, 0.1598)			**Train** 0.1646 **Valid** 0.1700 **Test** 0.1575		
Model	**Train** (0.9705, 0.9705, 0.9705) **Valid** (0.9670, 0.9670, 0.9670) **Test** (0.9650, 0.9650, 0.9650)			**Train** (0.9536, 0.9528, 0.9532) **Valid** (0.9479, 0.9519, 0.9499) **Test** (0.9387, 0.9400, 0.9394)			**Train** 0.9705 **Valid** 0.9670 **Test** 0.9650		

Results & Analysis
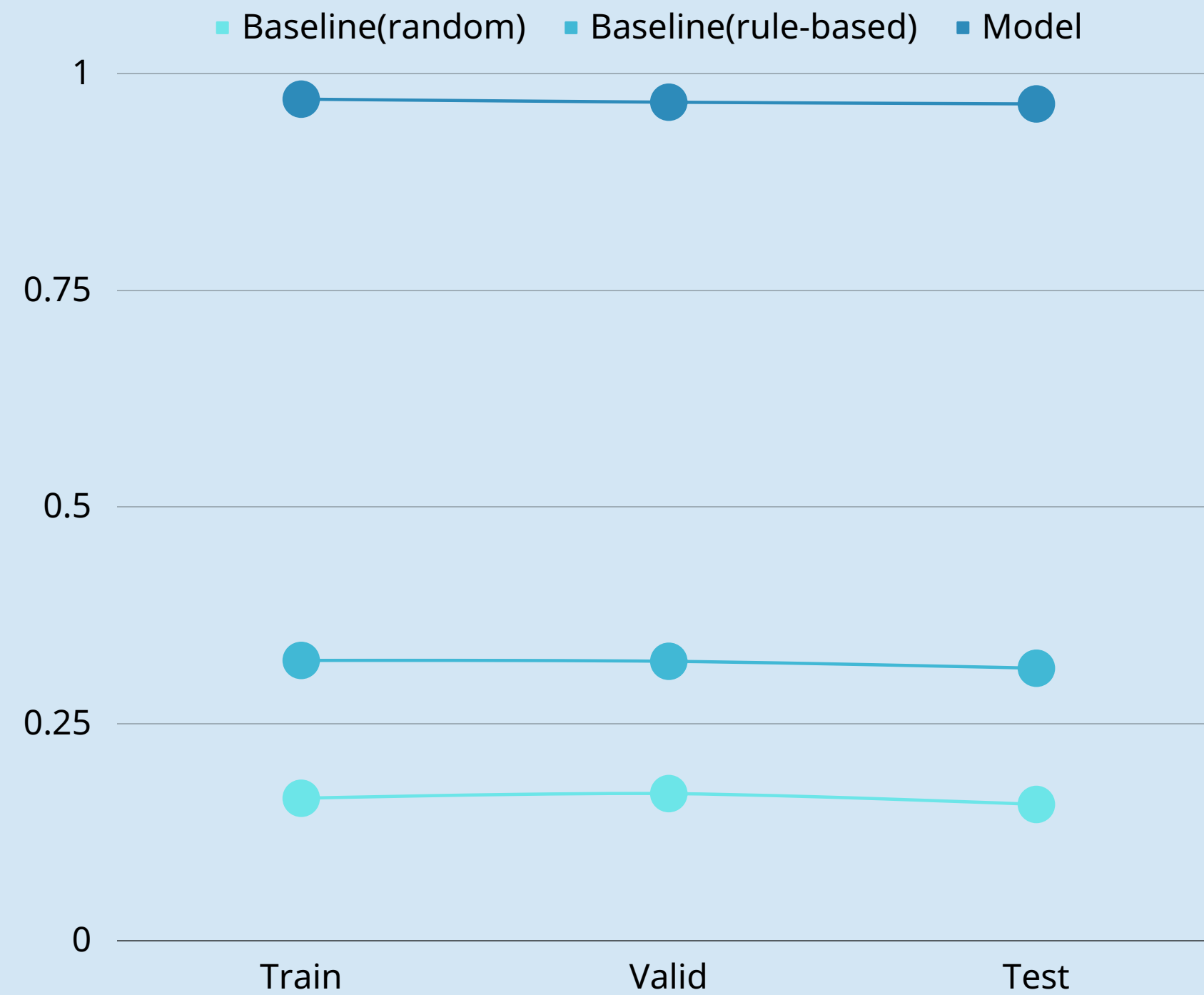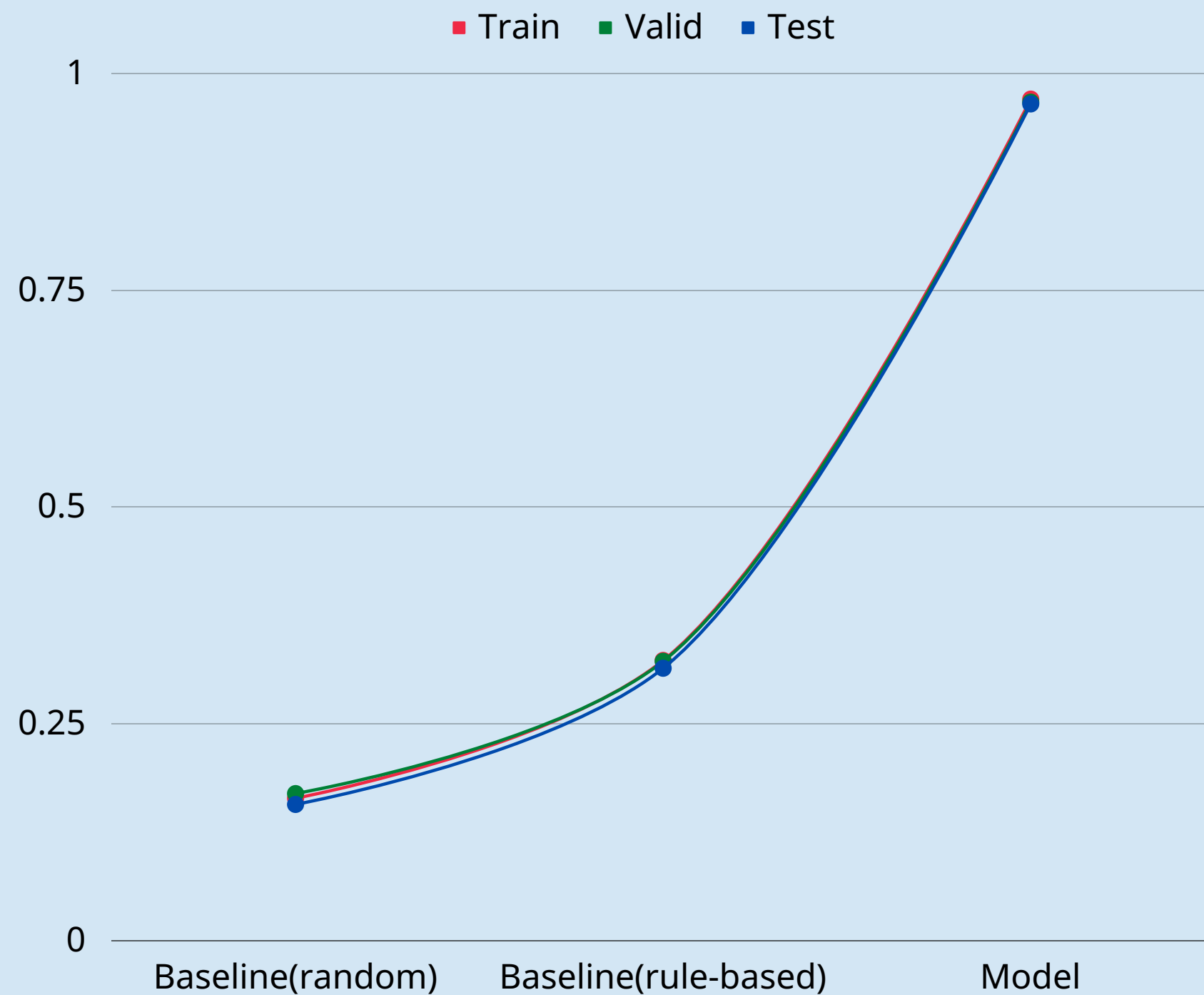
⚠️ Error Analysis

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?