Emotion Classification using Tweet

Group

1337

(Joe) Jiaheng Kang z5419190

(Dylan) Yuyang Xiao z5529467

Silu Yang z5526686

Tong Zhou z5418247

Xiao Dong z5571806

UNSW SYDNEY

01
Motivation
Problem Statement

(Joe) Jiaheng Kang
Silu Yang

02
Literature Review

(Joe) Jiaheng Kang
Tong Zhou
Silu Yang

03
Dataset/EDA

Xiao Dong

04
Methods

(Joe) Jiaheng Kang

05
Results

(Dylan) Yuyang Xiao

06
Discussion
Conclusion

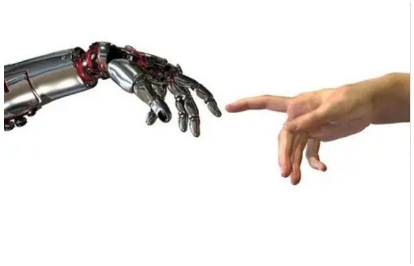
Tong Zhou

Motivation

Emotions

- anger
- fear
- joy
- love
- sadness
- surprise

AI MODELS



Human-Computer Interaction

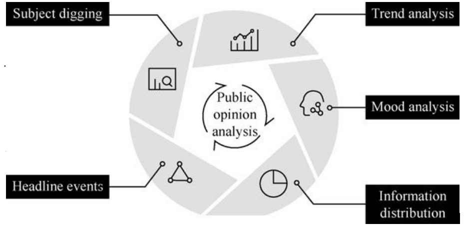
market research

Mental Health Monitoring

Customer Service

Public Opinion Analysis

Advertising



Problem Statement: Emotion Classification using Tweets

Background:
Tweets express emotions in subtle, varied ways, often shaped by personal and cultural context. This makes automatic recognition challenging, especially in informal online language.

Significance:
Accurate emotion detection in tweets supports public opinion analysis and mental health applications.

Key Challenge:
Capturing nuanced emotions is difficult—The main challenges are the highly imbalanced emotion distribution, the short and noisy nature of tweets, and the difficulty of accurately recognizing subtle emotions, especially for rare classes.

Our Aim:
Our aim is to compare CNN, BiLSTM, and advanced transformers (RoBERTa/DeBERTa) models to identify the most effective architecture for emotion classification in tweets. Based on the best-performing model, we further incorporate attention pooling and class balancing techniques to better capture subtle emotional cues and improve recognition of both common and rare emotions in real-world social media data.

Literature Review: Emotion Extraction and Classification from Twitter Text

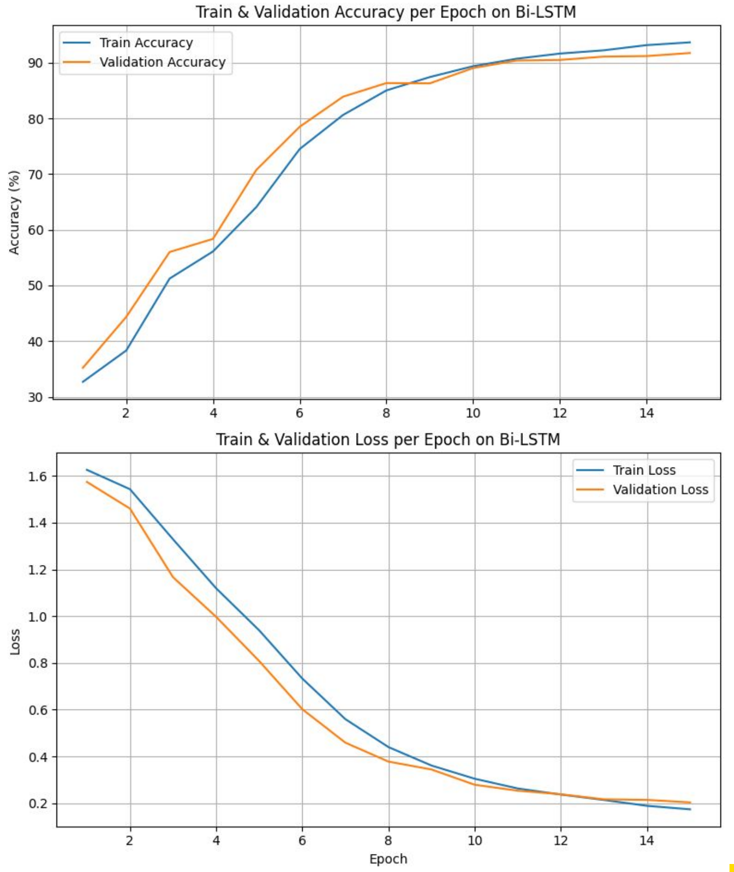
CNN:
Efficient at local feature extraction;
struggles with overall sentence context and complex emotions.

BiLSTM:
Captures global dependencies, outperforming CNNs on accuracy/F1;
training slower, more resource-intensive.

Limitations:
Depend on static word embeddings, lack adaptability to context.
Poor at handling class imbalance.
Architectural complexity increases cost with only minor performance gains.

Recent Advances:
Transformer models (RoBERTa, DeBERTa) provide stronger contextual understanding and improved solutions for class imbalance, setting new standards in emotion classification.

*************************	*************************** Val Max ***************************	
Max Val Accuracy: 90.60%	Averag Val Accuracy: 91.75%	Averag Train Accuracy on Bi-LSTM: 74.719
Max Val Precision: 90.80%	Averag Val Precision: 100.00%	Averag Val Accuracy on Bi-LSTM: 76.247
Max Val Recall: 90.60%	Averag Val Recall: 91.75%	Averag Train Loss on Bi-LSTM: 0.669
Max Val F1 Score: 90.65%	Averag Val F1 Score: 91.77%	Averag Val Loss on Bi-LSTM: 0.613
*************************	*************************** Test ***************************	
Test Accuracy: 89.30%	Test Accuracy: 90.95%	
Test Precision: 89.32%	Test Precision: 90.91%	
Test Recall: 89.30%	Test Recall: 90.95%	
Test F1 Score: 89.26%	Test F1 Score: 90.91%	



Presenter: Silu Yang (z5526686)

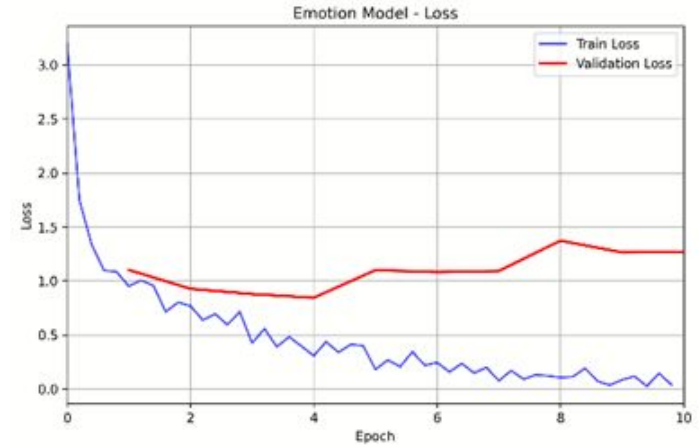
Literature Review

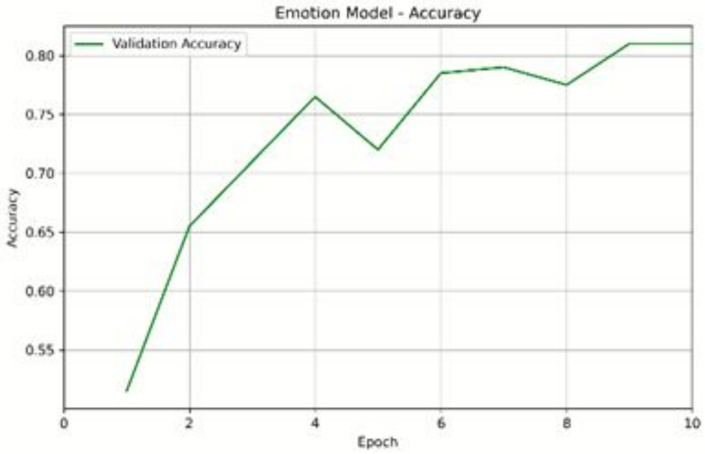
FEEL-IT: Emotion and Sentiment Classification for the Italian Language
ACL 2021 Workshop

Key Features:
- Model constructed using high quality Italian Tweets (from COMMOWNCRAW ITA)
- UmBERTo-FT Model obtained the best result in terms of overall performance
- Trained to perform both sentiment and emotion classification

Modifitions to the FEEL-IT Model to enable prediction of english tweets:
- Map dair-ai/emotion dataset's ("Love", "Surprise" -> "Joy") to FEEL-IT Format (anger, fear, joy, sadeness)
- Trained a model with the pretrained FEEL-IT model and the dair-ai/emotion dataset.

Primary Issue with the model:
- The emotions are annotated by Italian speakers, which reflects Italian cultural and linguistic norms. Even after retraining with English data, these annotations still fail to capture English culture, syntax, and context effectively.





Literature Review

CARER: Contextualized Affect Representations for Emotion Recognition
(EMNLP 2018, Saravia et al.)

Key Features:

- Construct an "emotion graph."
- Generate semantic patterns.
- Weight patterns per emotion.
- Classify with a CNN (CARER).

Explanation of Emotional Graph:

Ugh, he forgot my birthday again—so mad right now!

"mad", "ugh" contains high emotions

- Emotional word (graph) = Subjective tweet - Objective tweet)

The pattern of (connector words, emotional words) e.g (so mad) resolves the issue of identifying "sooo maaaaad" as the model can group the two words "sooo" and "maaaaad" together for classification

Issues:

- Did not resolve class imbalance
- Hard to implement and high costs

Dataset

Source: [dair-ai/emotion · Datasets at Hugging Face]

Text

- The text field contains short emotional sentences from English Twitter posts
- Text lengths vary significantly, ranging from 7 to nearly 300 characters

Label

- The training set includes 16,000 samples
- The dataset defines 6 basic emotion classes

Search this dataset

text string · *lengths* 7 300	label class label 6 classes
i didnt feel humiliated	0 sadness
i can go from feeling so hopeless to so damned hopeful just from being around someone who cares and is awake	0 sadness
im grabbing a minute to post i feel greedy wrong	3 anger
i am ever feeling nostalgic about the fireplace i will know that it is still on the property	2 love
i am feeling grouchy	3 anger
ive been feeling a little burdened lately wasnt sure why that was	0 sadness
ive been taking or milligrams or times recommended amount and ive fallen asleep a lot faster but i also feel like so funny	5 surprise
i feel as confused about life as a teenager or as jaded as a year old man	4 fear
i have been with petronas for years i feel that petronas has performed well and made a huge profit	1 joy
i feel romantic too	2 love
i feel like i have to make the suffering i m seeing mean something	0 sadness
i do feel that running is a divine experience and that i can expect to have some type of spiritual encounter	1 joy
i think it s the easiest time of year to feel dissatisfied	3 anger
i feel low energy i m just thirsty	0 sadness
i have immense sympathy with the general point but as a possible proto writer trying to find time to write in the corners of life and with no sign of an agent let alone a publishing contract this feels a little precious	1 joy
i do not feel reassured anxiety is on each side	1 joy
i didnt really feel that embarrassed	0 sadness
i feel pretty pathetic most of the time	0 sadness

‹ Previous 1 2 3 ... 160 Next ›

Data Analysis

01 Dataset Overview

Provided a structural summary of the dataset, including data types, missing values, and basic statistics.

02 Class Distribution

Analysed the emotion class counts and identified class imbalance

03 Text Length Stats

Computed text length statistics to assess variability in input size

04 VIF

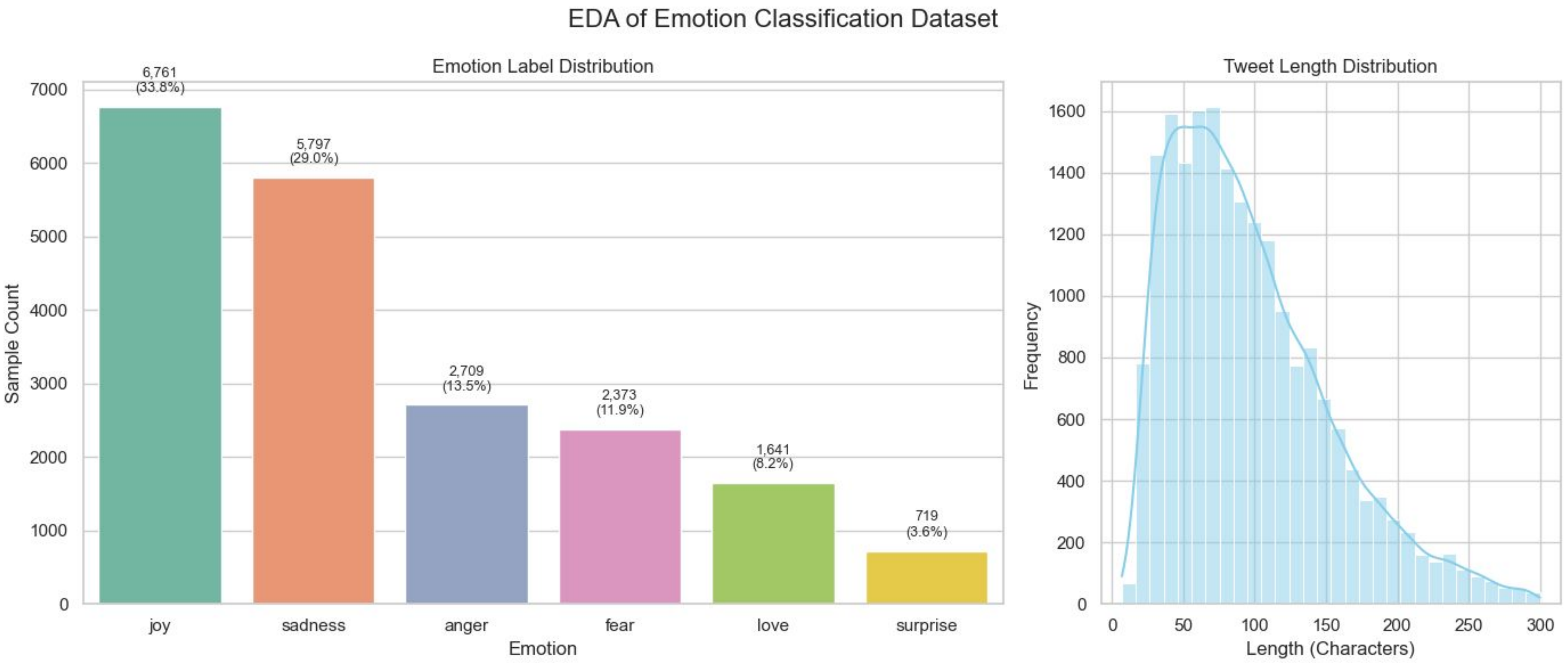
Conducted Variance Inflation Factor analysis to check for multicollinearity

05 Data Leakage

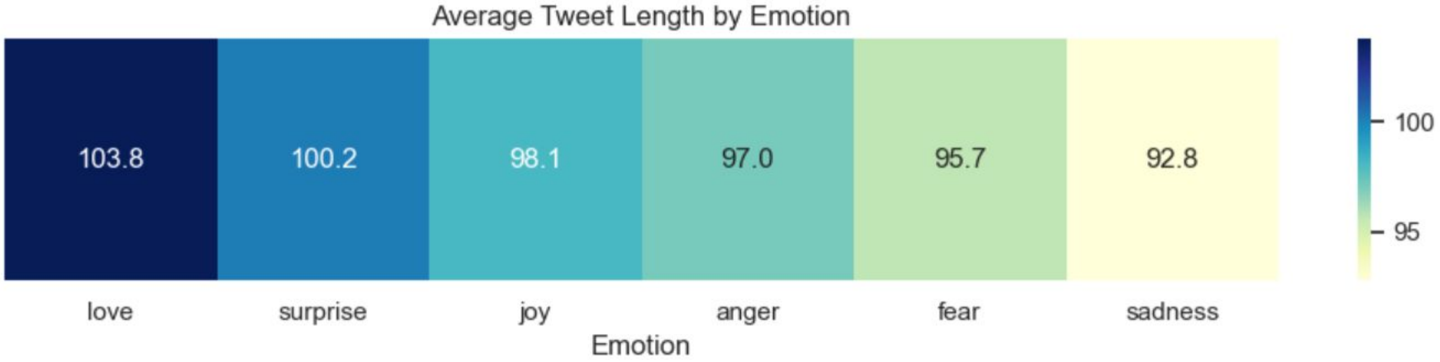
Test the predictive power of text length alone, to find leakage.

06 Conclusion

Class imbalance exists and must be addressed before modeling.



Presenter: XIAO DONG(z5571806)



Section	Metric	Value
Class Distribution	Max/Min Imbalance Ratio	9.40x
Class Distribution	Mean	96.67
Text Length Stats	Std	55.78
Text Length Stats	Min	129
Max	Max	129
Leakage Conclusion	No significant	0.3380

Methods

01 Reuse CARER's cleaned data pipeline / preproceed data

02 Utilize a DeBERTa model to do the final perdiction

03 Introduce undersampling(repeated sampling) and oversampling

Methods

- Alternative choices:
- RoBERTa vs DeBERTa
  - DeBERTa’s disentangled attention better captures subtle semantic clues (like sarcasm, emojis, or indirect emotion), which is critical in emotion classification.
  - Relative positional encoding (which can understand the context better)
  - In GoEmotions (27-label fine-grained emotion dataset), DeBERTa shows higher F1 scores than RoBERTa.
  - DeBERTa typically converges faster

Methods

 Alternative choices:

 Oversampling (SMOTE or similar)

- Tokenisation
- Non-continuous data
- Unreliable synthesised data

Examples:

- I want to kill! (anger)
- I hate you! (anger)

- want to I you (nonsense)
- I hate to kill (anger?)

Results

The model achieves a macro F1 score of **92%**, loss of **0.22** and accuracy of **93%** on the test set.
Compared to baseline model, macro F1 score of **86%**, loss of **0.30** and accuracy of **92%** on the test set.

```
Epoch 9: 100%  250/250 [04:51<00:00, 0.86it/s, v_num=2, train_loss_step=0.000158, train_acc_step=1.000, val_loss_step=0.352, val_acc_step=0.875, val_loss_epoch=0.222, val_acc_epoch=0.937, val_f1=0.907, train_loss_epoch=0.0084, train_acc_epoch=0.997, train_f1=0.993]

INFO:pytorch_lightning.utilities.rank_zero:`Trainer.fit` stopped: `max_epochs=10` reached.
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES: [0]
/usr/local/lib/python3.11/dist-packages/pytorch_lightning/trainer/connectors/data_connector.py:425: The 'test_dataloader' does not have many workers which may be a bottleneck. Consider increasing the value of the `num_worke
Testing DataLoader 0: 100%  32/32 [00:12<00:00, 2.47it/s]
```

Test metric	DataLoader 0
test_acc_epoch	0.9244999885559082
test_f1	0.8626927137374878
test_loss_epoch	0.30245256423950195

```
[{'test_loss_epoch': 0.30245256423950195,
  'test_acc_epoch': 0.9244999885559082,
  'test_f1': 0.8626927137374878}]
```

```
Epoch 6: 100%  188/188 [00:37<00:00, 4.95it/s, v_num=2, train_loss=0.0499, train_acc=1.000, train_f1=1.000, val_loss=0.252, val_acc=0.924, val_f1=0.917]
INFO:pytorch_lightning.callbacks.early_stopping:Metric val_f1 improved. New best score: 0.877
INFO:pytorch_lightning.callbacks.early_stopping:Metric val_f1 improved by 0.020 >= min_delta = 0.0. New best score: 0.897
INFO:pytorch_lightning.callbacks.early_stopping:Metric val_f1 improved by 0.021 >= min_delta = 0.0. New best score: 0.917
INFO:pytorch_lightning.callbacks.early_stopping:Metric val_f1 improved by 0.007 >= min_delta = 0.0. New best score: 0.924
INFO:pytorch_lightning.callbacks.early_stopping:Monitored metric val_f1 did not improve in the last 3 records. Best score: 0.924. Signaling Trainer to stop.
INFO:pytorch_lightning.accelerators.cuda:LOCAL_RANK: 0 - CUDA_VISIBLE_DEVICES: [0]
Testing DataLoader 0: 100%  188/188 [00:06<00:00, 29.86it/s]
```

Test metric	DataLoader 0
test_acc	0.9336666464805603
test_f1	0.9260219931602478
test_loss	0.21613095700740814

```
[{'test_loss': 0.21613095700740814,
  'test_acc': 0.9336666464805603,
  'test_f1': 0.9260219931602478}]
```

Results(acc &loss)

01

Train acc

02

Test acc

03

Val acc

04

Train loss

05

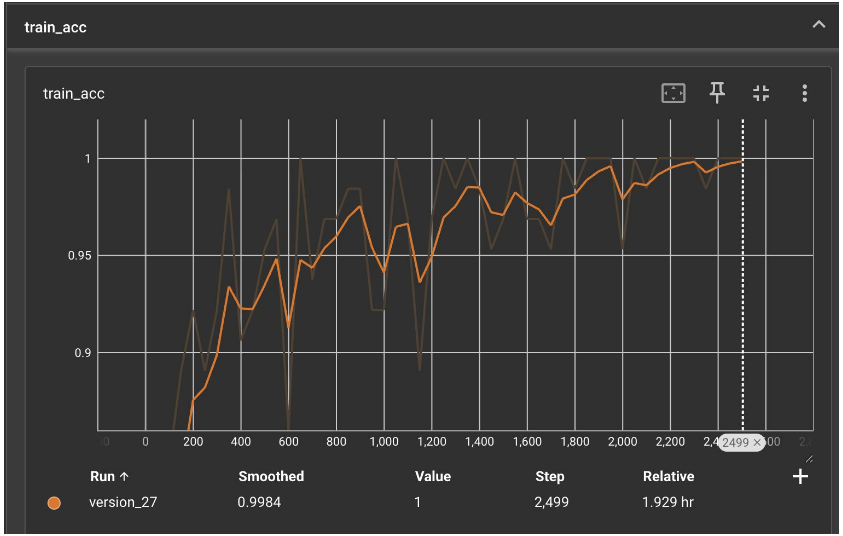
Test loss

06

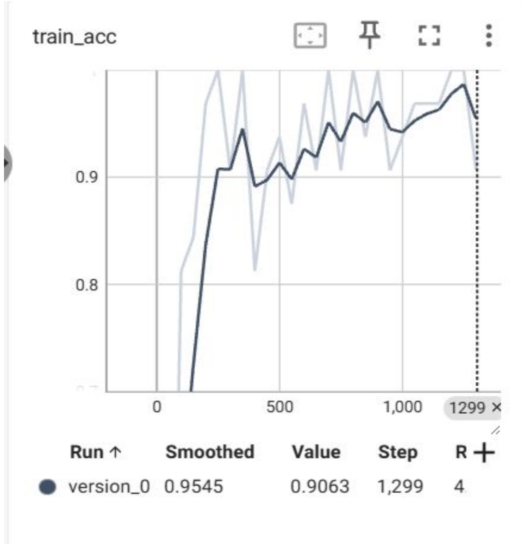
Val loss

Train acc

Baseline: 0.9984

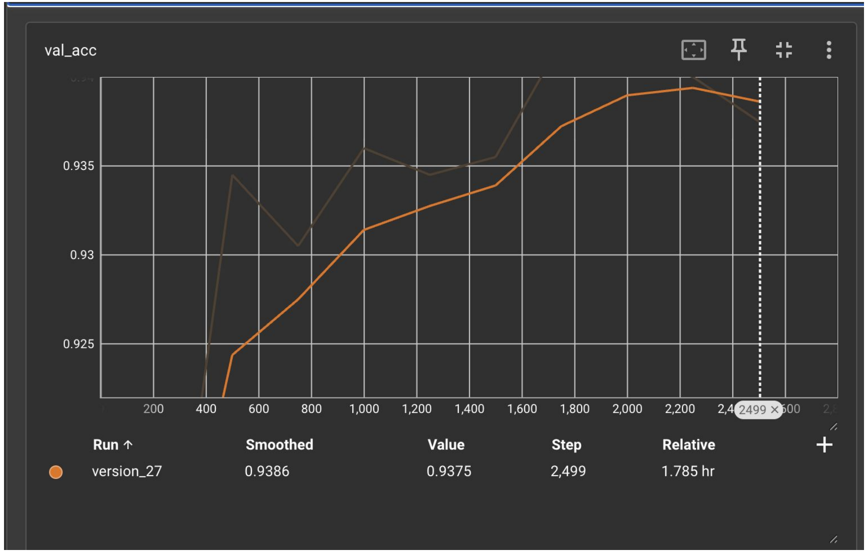


Final:0.9545

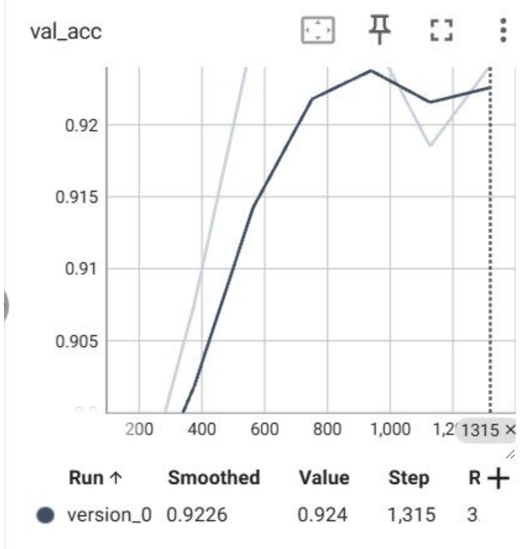


Val acc

Baseline: 0.9386

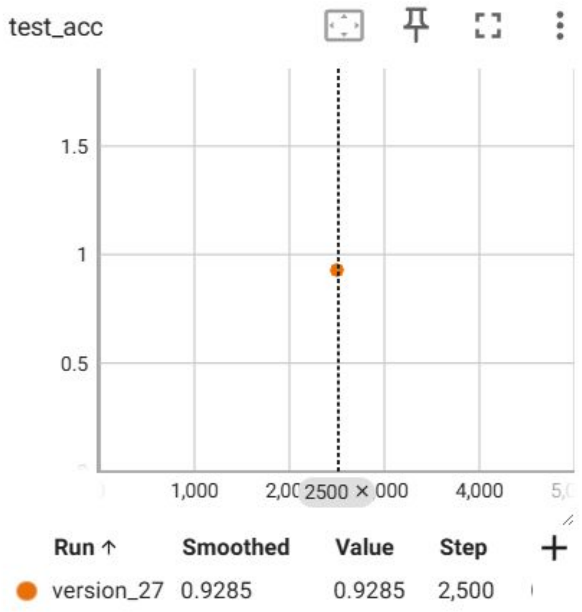


Final: 0.9226

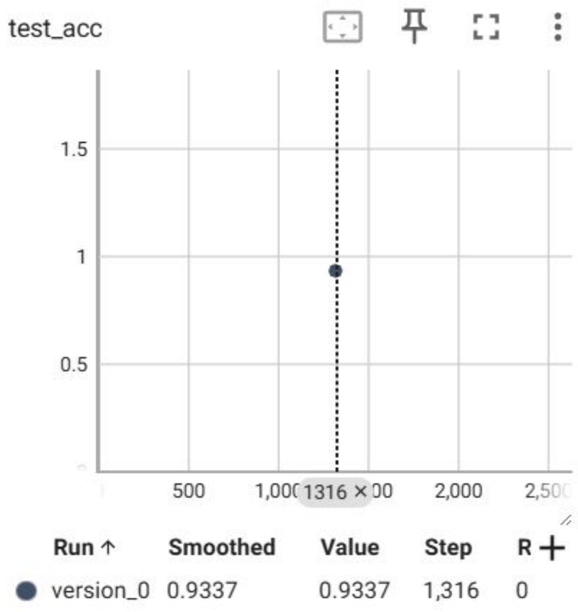


Test acc

Baseline: 0.9285



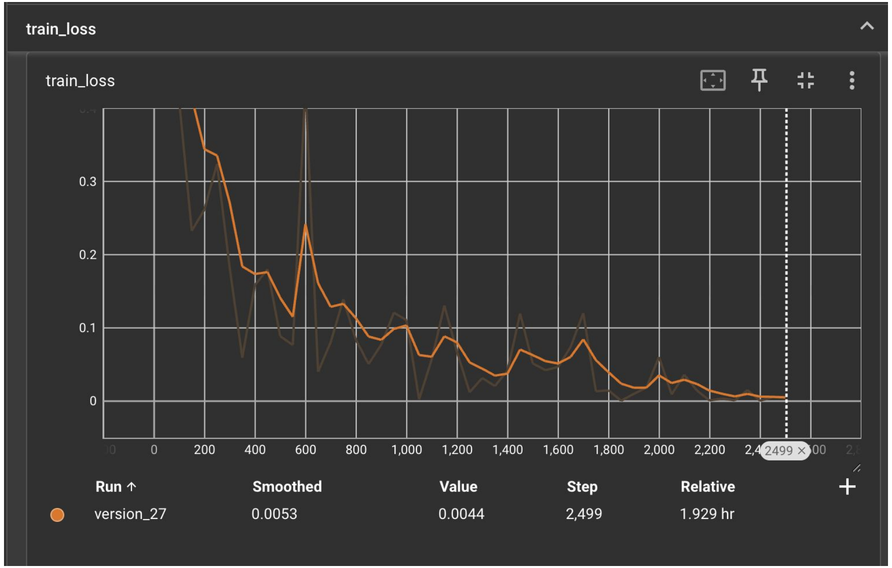
Final: 0.9337

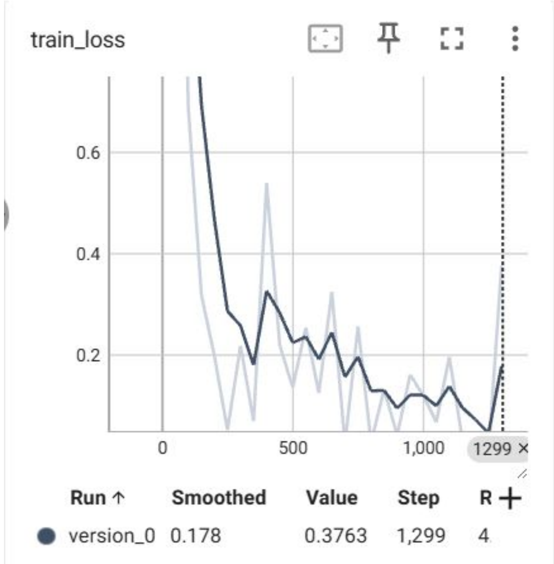


Train loss

Baseline:0.0053

Final: 0.3763

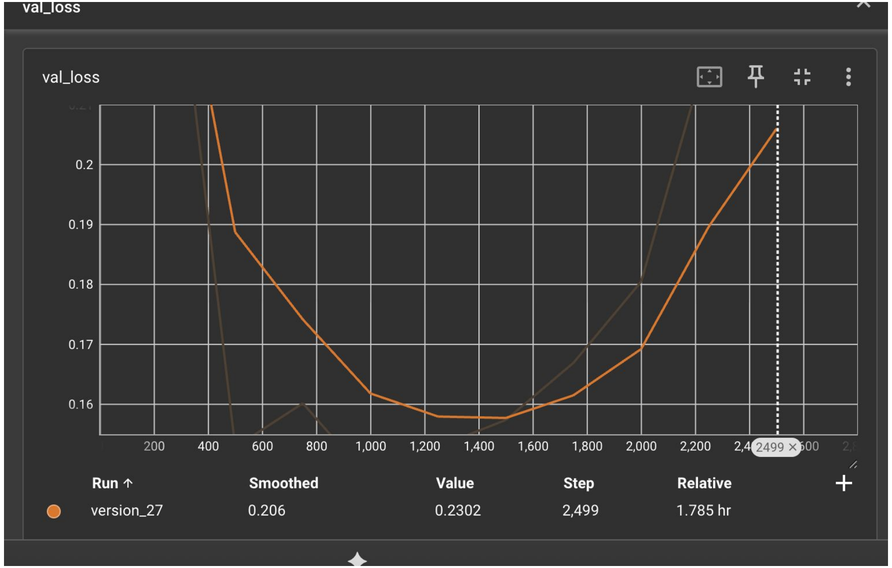


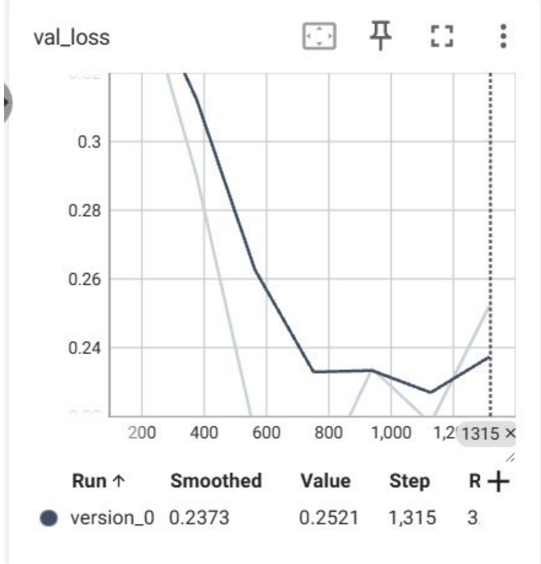


Val loss

Baseline:0.206

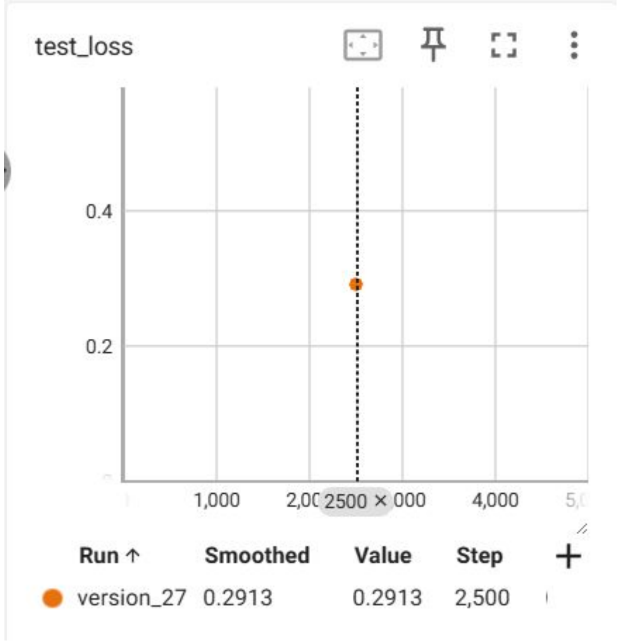
Final: 0.2373



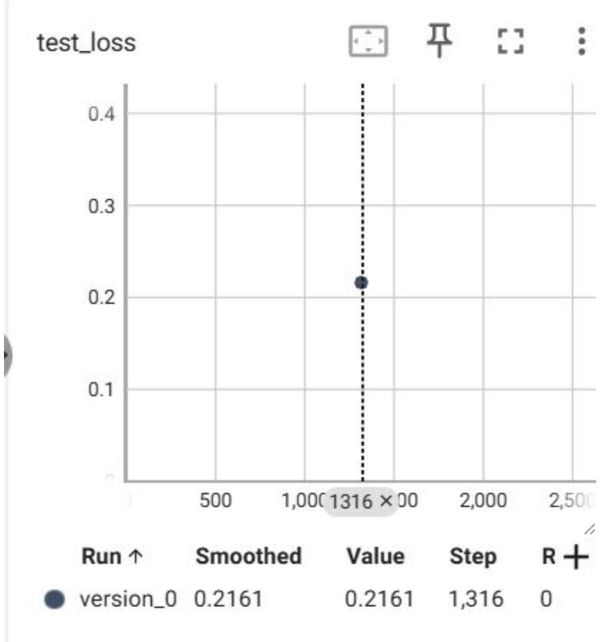


Test loss

Baseline: 0.2913



Final: 0.2161



Discussion

New Model	Baseline
~143m Parameters	~82m Parameters
Trains Fasters (30minutes in Google Colab) due to optimisation made to account for pytorch performance	Trains Slower (2-3 Hours in Google Colab)
Better Accuracy (93%) and F1 Score (92%)	Worth Accuracy (92%) and F1 Score (86%)

Key Strength

- The significant improvement in F1 Score shows the effectiveness of balancing data
- The utilisation of DeBERTa also contributed to the improve in accuracy and training time

Discussion (continued)

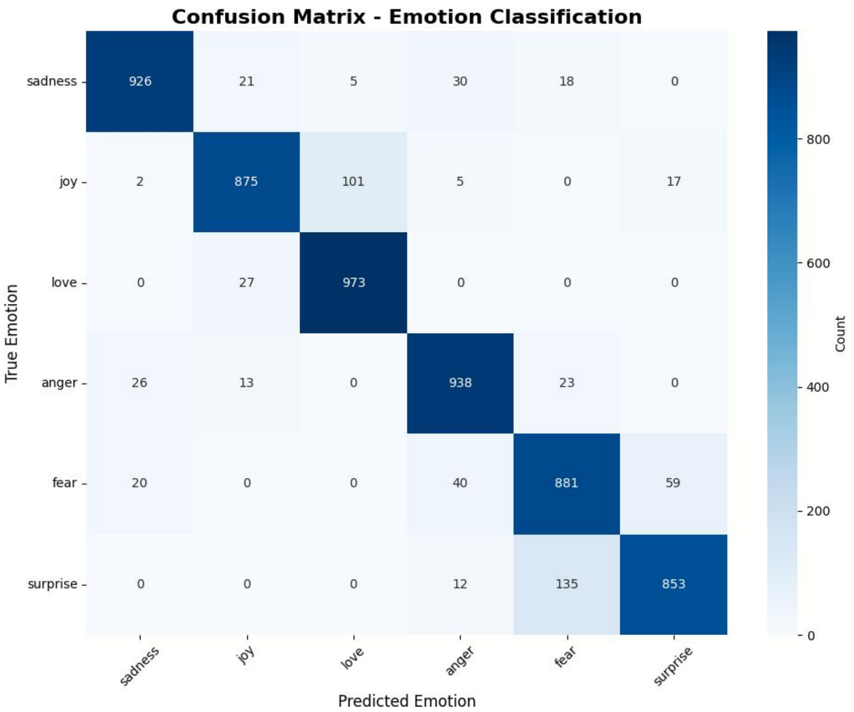
Weakness & Limitation:

The Model struggles to distinguish between

- Joy vs Love
- Fear vs Suprise

Future Work:

- Exploration of DeBERTa model with larger number of parameters.
- Explore multi-label classification models to capture co-existing emotions.



Conclusion

Through

- Changing system archiact to DeBERTa model
- Performing oversampling and undersamplying to balance the tweet data.

We

- Improved Performance: Accuracy and F1 Score
- Tackled class imbalance issues
- Reduced Computational Cost and Training Time

However, model are still limited to

- classification of close emotions (joy vs love, fear vs surprise)
- the number of parameters

In which we are looking to explore further in the future!


Questions