MOOD CLASSIFICATION FOR SONG LYRICS AND POETRY

Meet the Team

Lia Berman

- Gathered, cleaned, and organized datasets
- Organized the project deliverables so the final presentation is cohesive
- Identified supporting background information and a relevant article for context.

Wilson Sun

- Testing different training method

Zeke Delaughter

- Trained baseline model
- Gathered data for training and testing

Kee Chee Pheng

- Created the script to get the probability distibutions of mood for song lyrics obtained from Spotify Million Song Dataset using pretrained mrm8488/t5-base-finetuned-emotion from Huggingface Library

AGENDA

01 Why it Matters

02 Datasets

03 Our Approach

04 Analysis

05 Future Work

06 Conclusion

WHY MOOD CLASSIFICATION MATTERS

01

The Research Gap

- Basic sentiment analysis is too limited.
- Creative text (lyrics/poems) expresses multiple emotions at once.
- Modern NLP needs models built specifically for emotion understanding.

02

Real-World Impact

- Streaming platforms use inconsistent mood labels.
- Better emotion detection improves recommendations and user experience.
- Useful for music therapy, mental-health tools, and creative analytics.

03

Technical Difficulty

- Lyrics and poems use metaphor and mixed emotions.
- Baseline models struggle, better models are needed.
- Domain specific testing

source: Mohammad (2021),Huang et al. (2023),From Joy to Fear” (2025)

DATASETS

kaggle

Link 1: 500 songs with Emotion Scores (MOS aggregated from 6 annotators)

Used: In the Testing

Link: link 1

Hugging Face

Link 2: social media text (specifically Twitter posts) where each text is labeled with one of six basic emotions: anger, fear, joy, love, sadness, surprise.

Used: Training

Link: link2

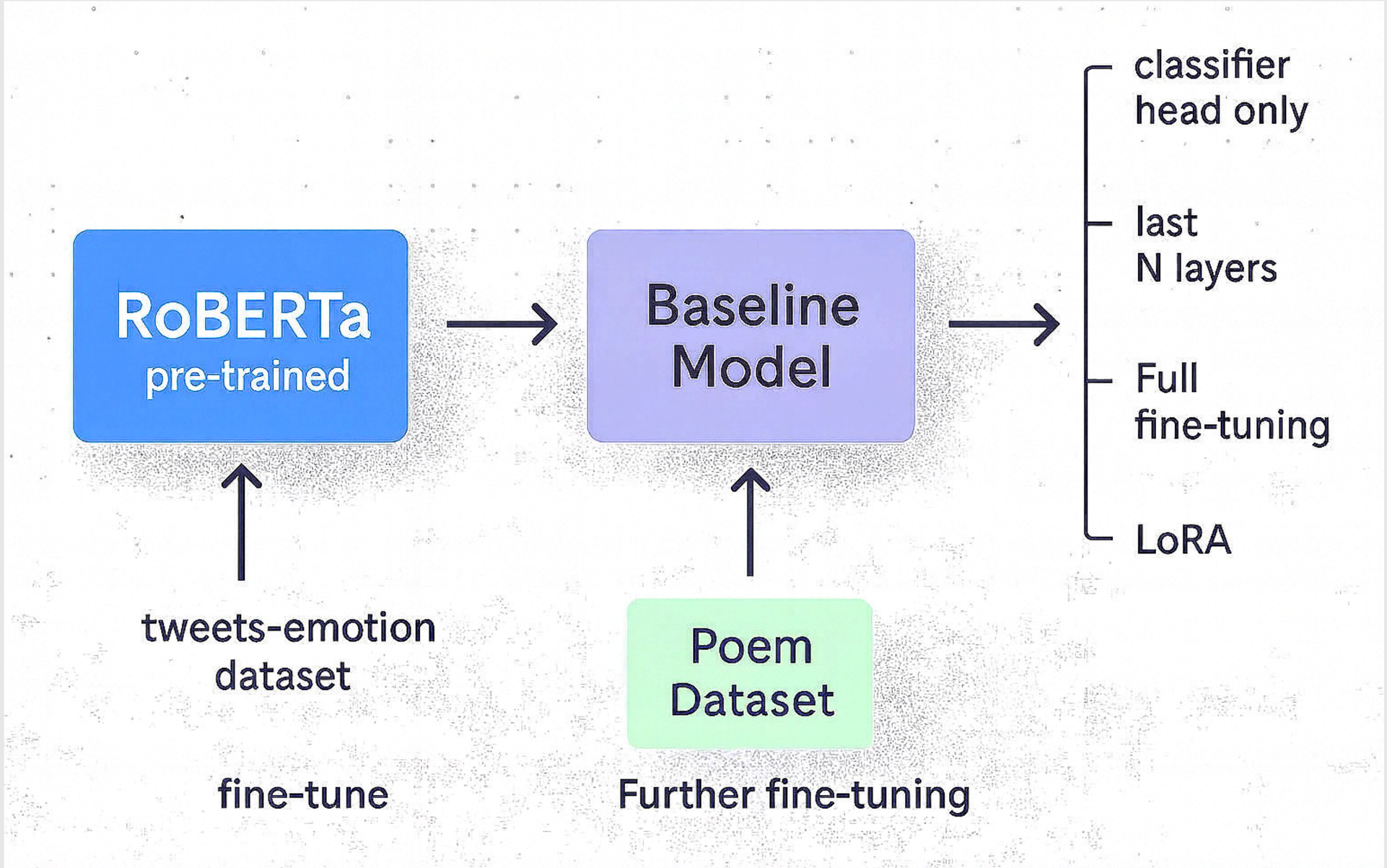
Mendeley Data

Link 3 The poems are labeled u with nine emotions: Love, Sad, Anger, Hate, Fear, Surprise, Courage, Joy, Peace. The corpus includes poems from 1850–2016, collected from the web and evaluated by human experts.

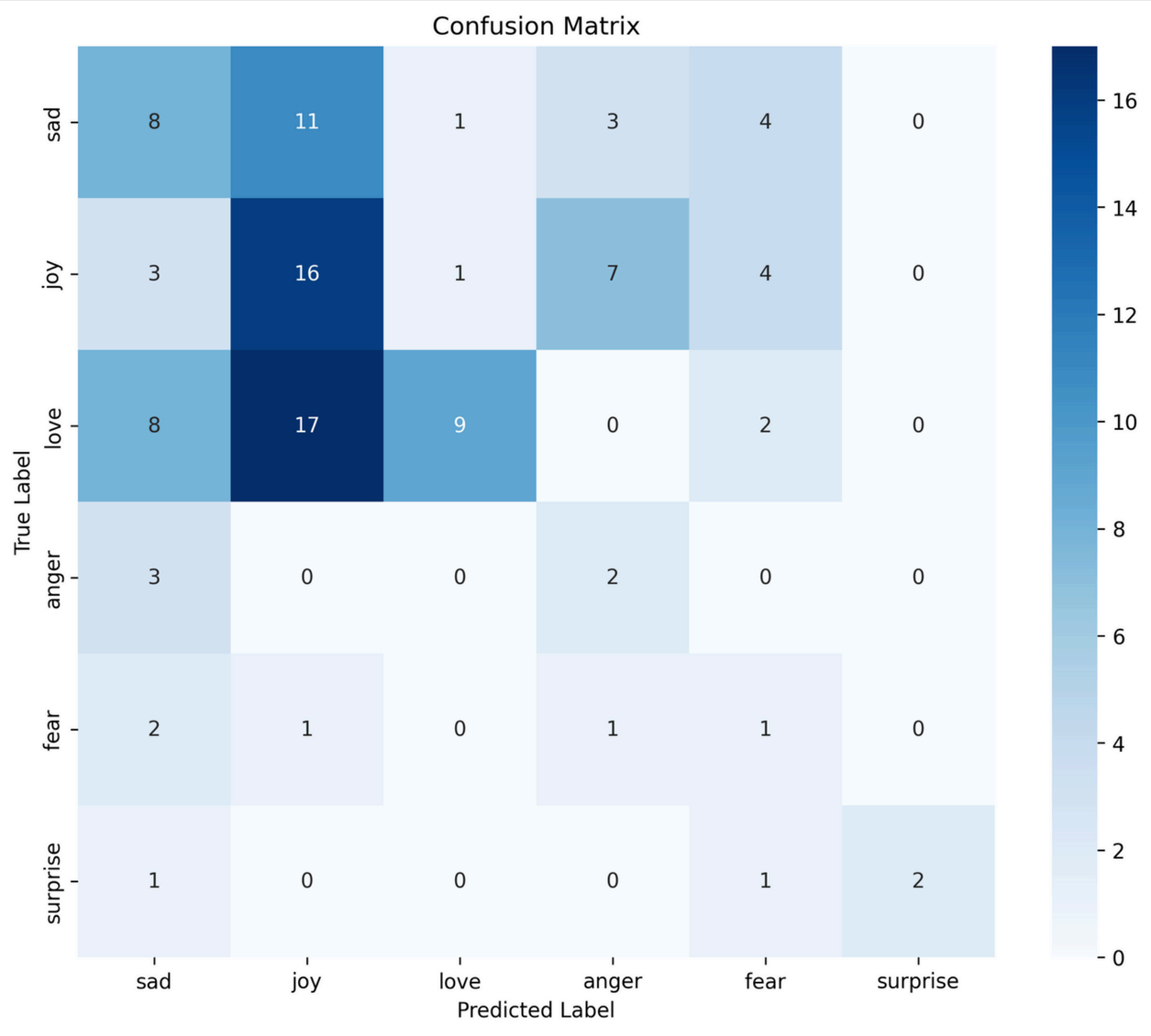
Used: Testing and Training

Link: link3

OUR APPROACH



BASELINE



Pre-trained **RoBERTa**

(encoder-only architecture naturally designed for discriminative tasks like sentiment classification)

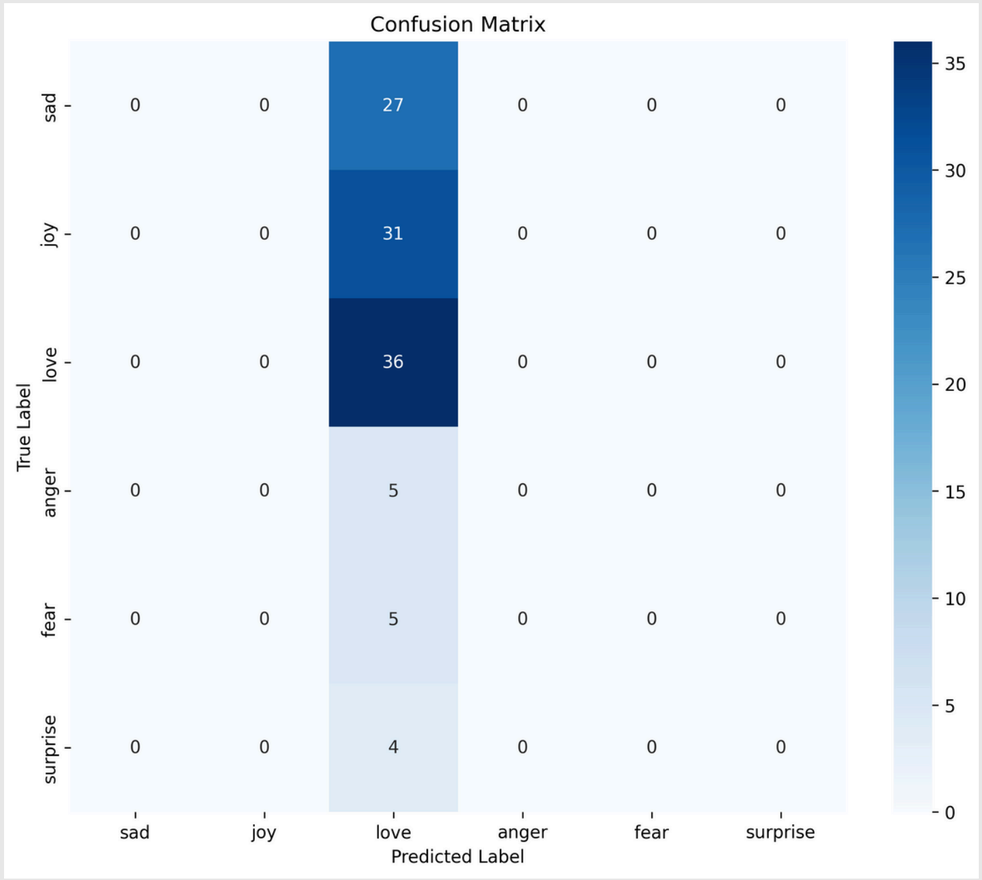
Fine-tuned with general texts (Social Media)

weighted avg f1 (poem): 0.37

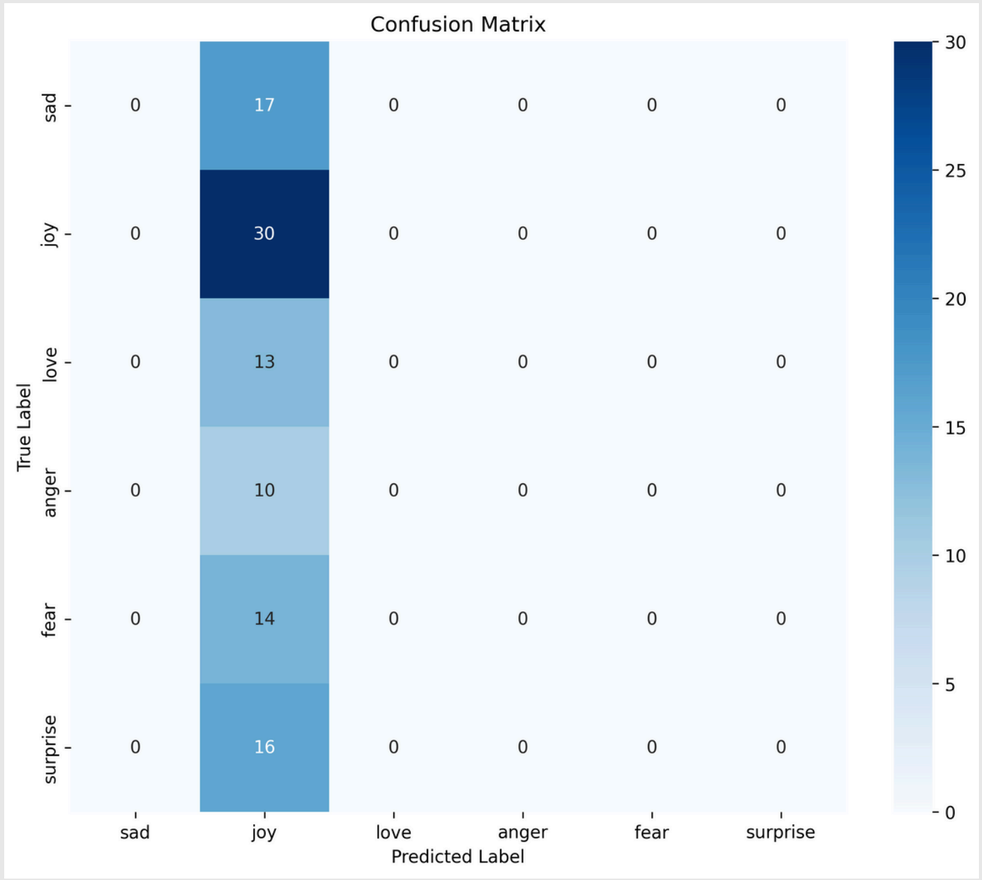
weighted avg f1 (lyrics): 0.17

DUMB BASELINE

poem weighted avg f1: 0.17

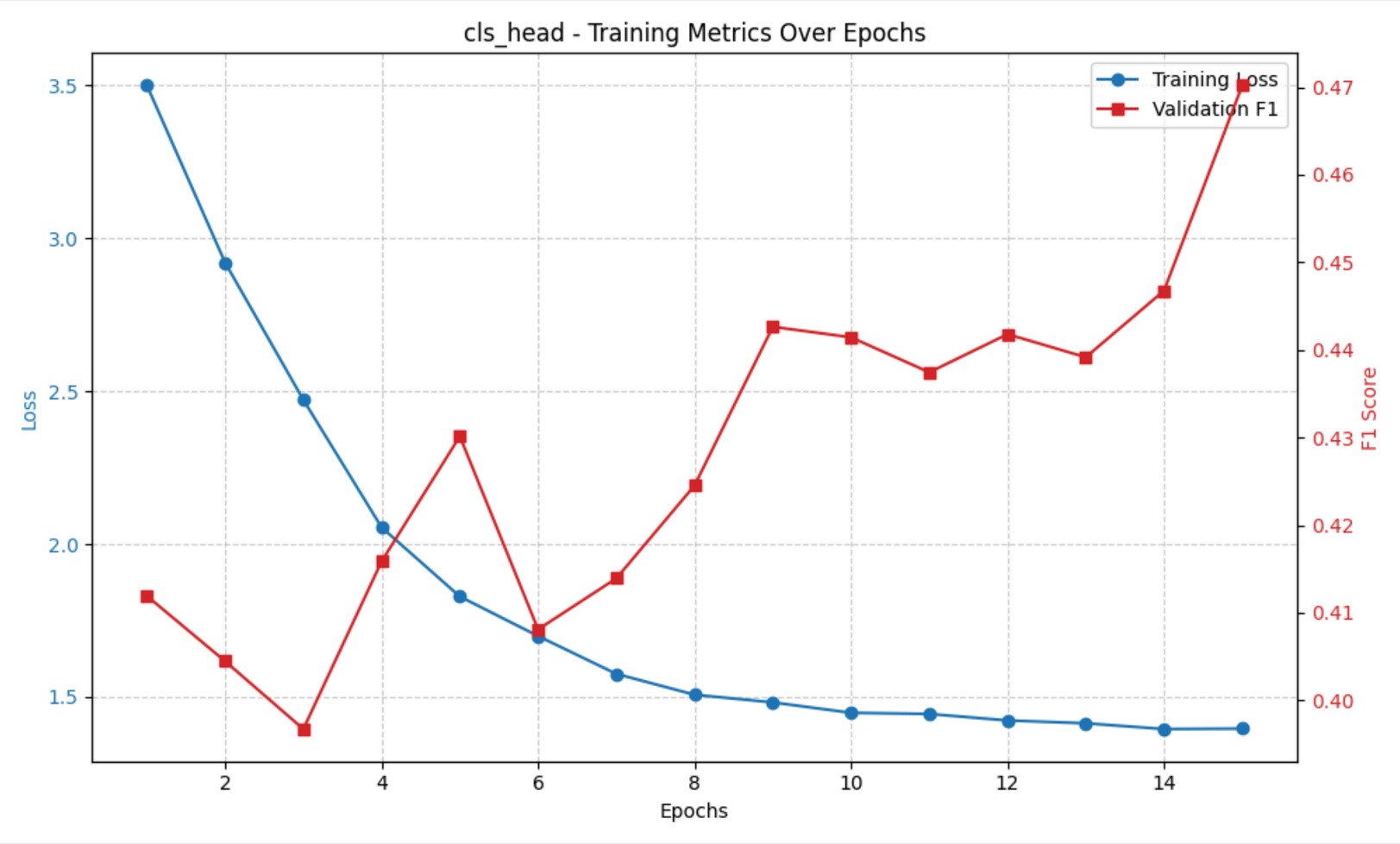


lyrics weighted avg f1: 0.14



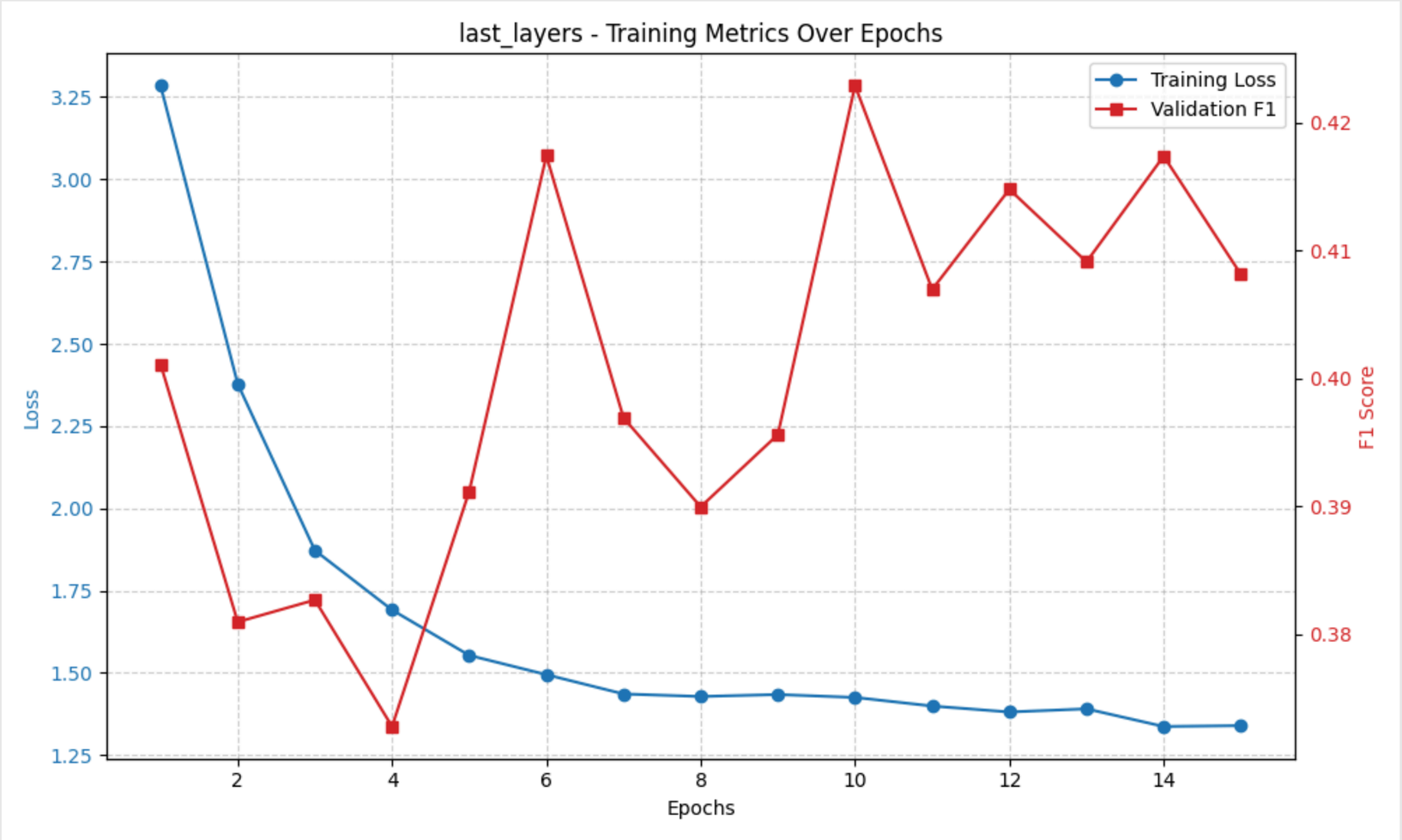
CLS HEAD

Only train the classification head on top of baseline



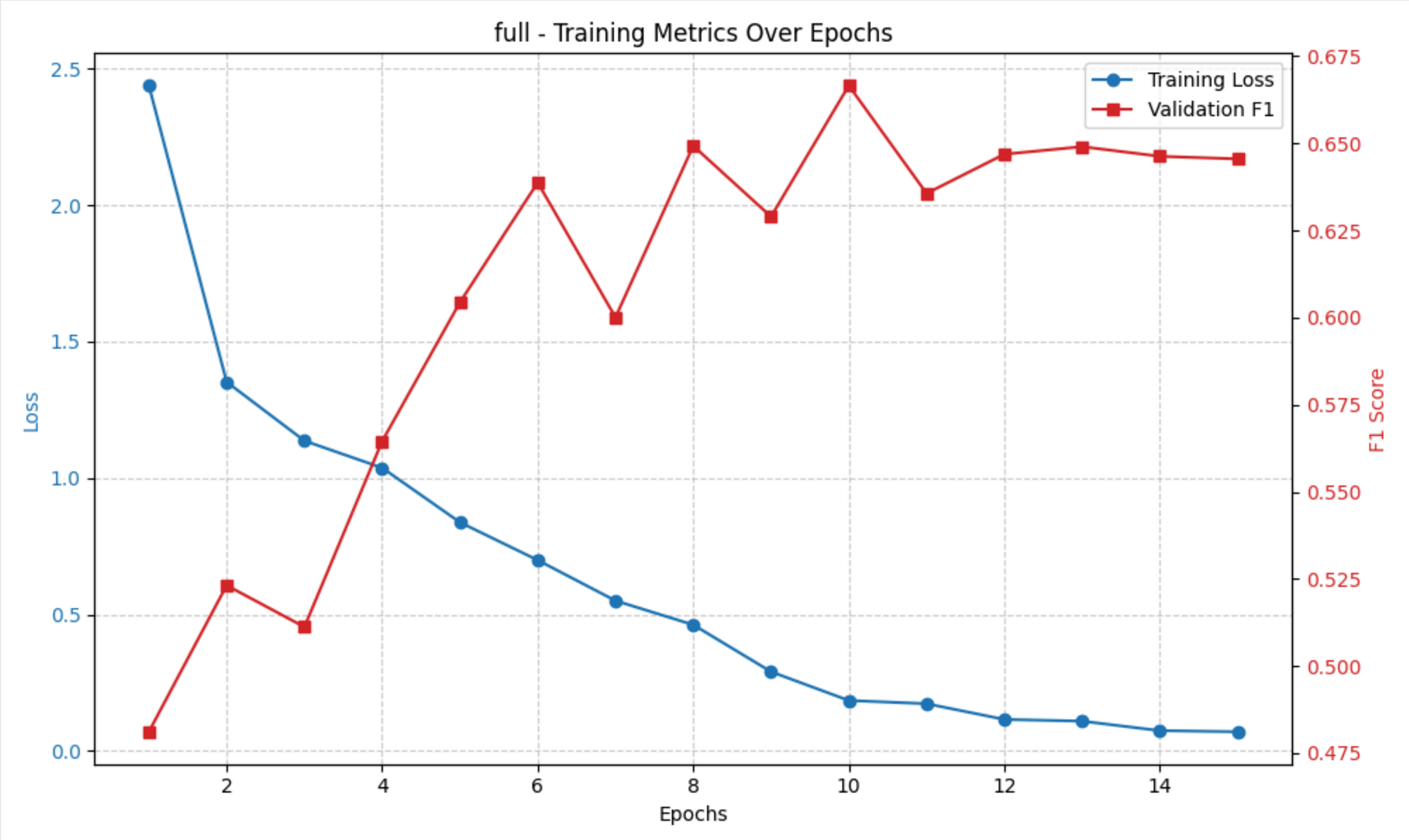
LAST N LAYERS

train the classification head and last N(2) layers of the transformer



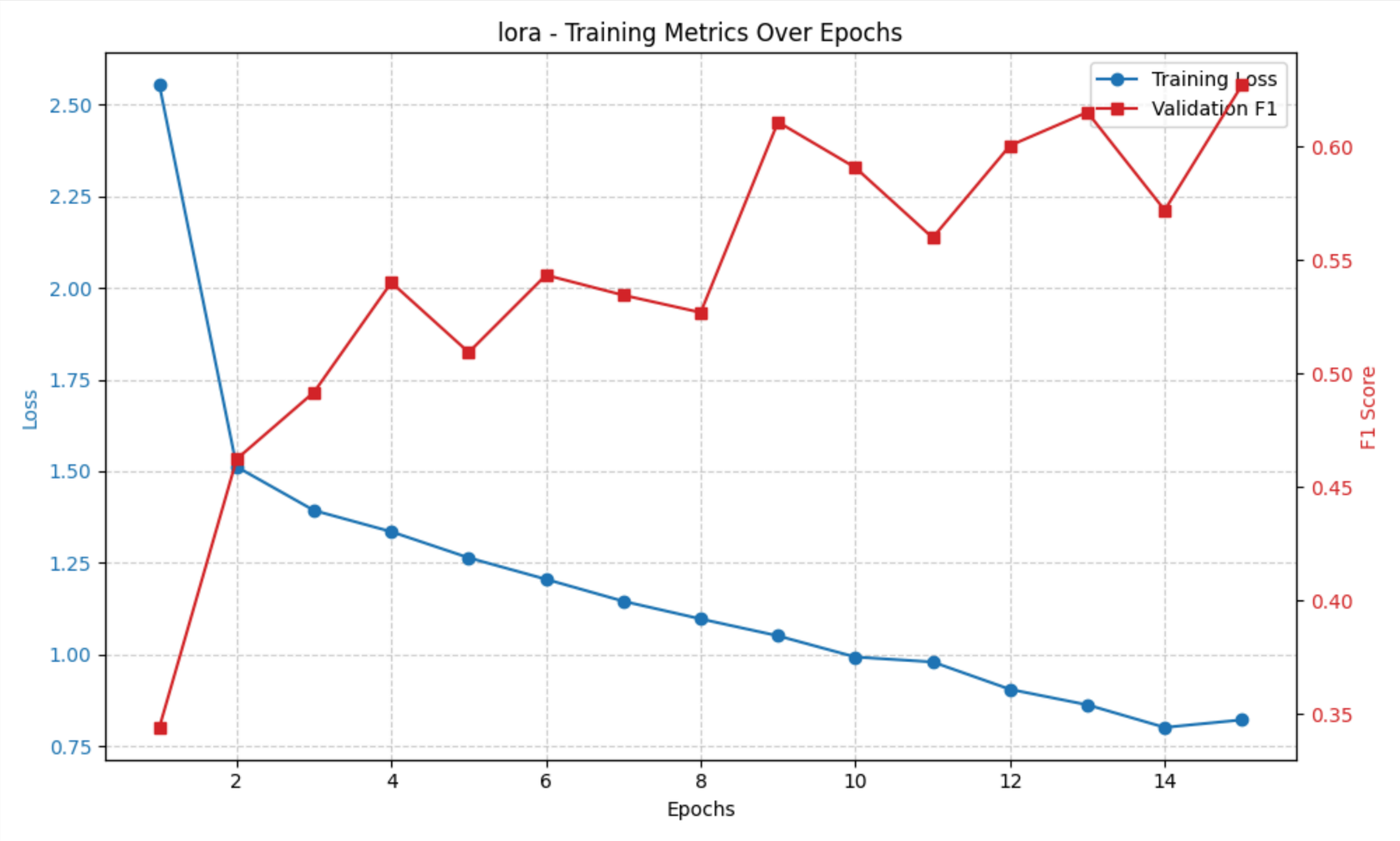
FULL

train every parameters

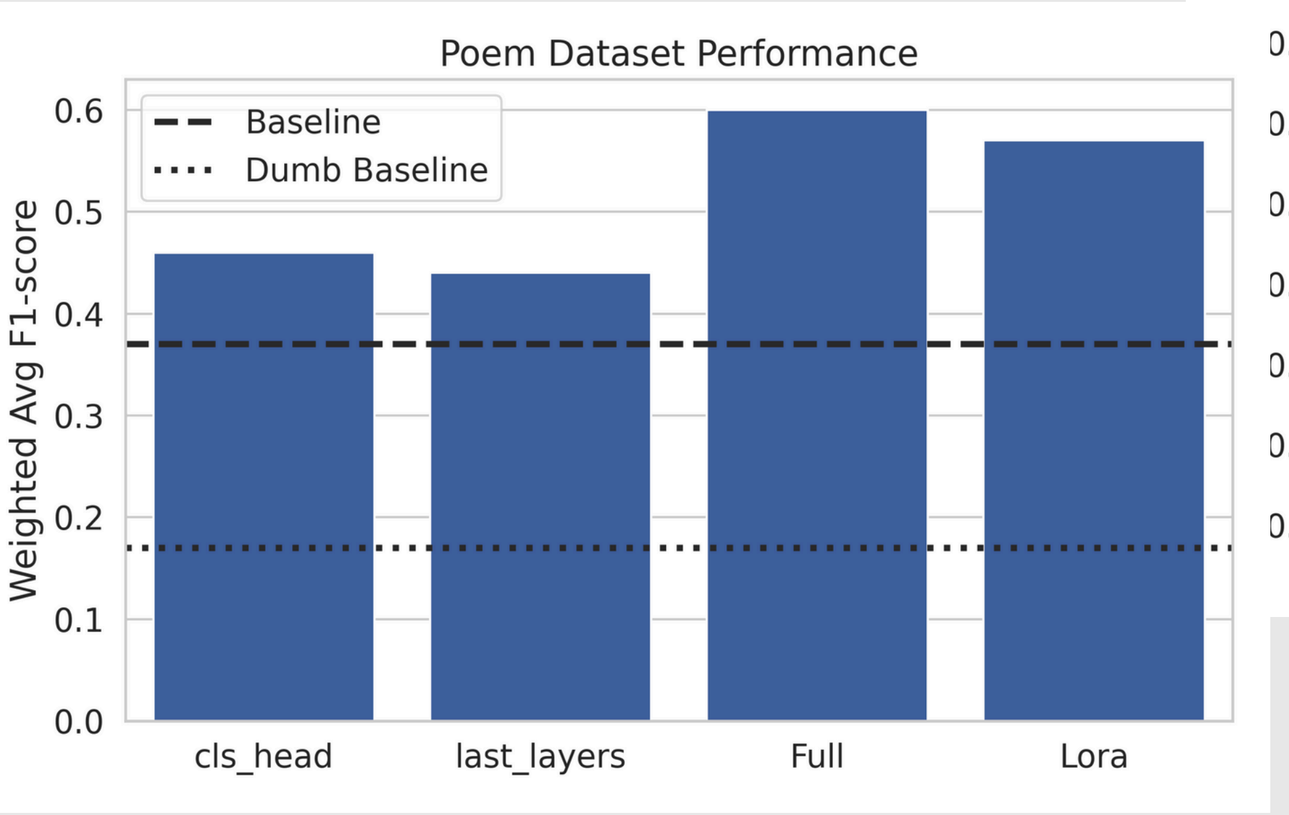


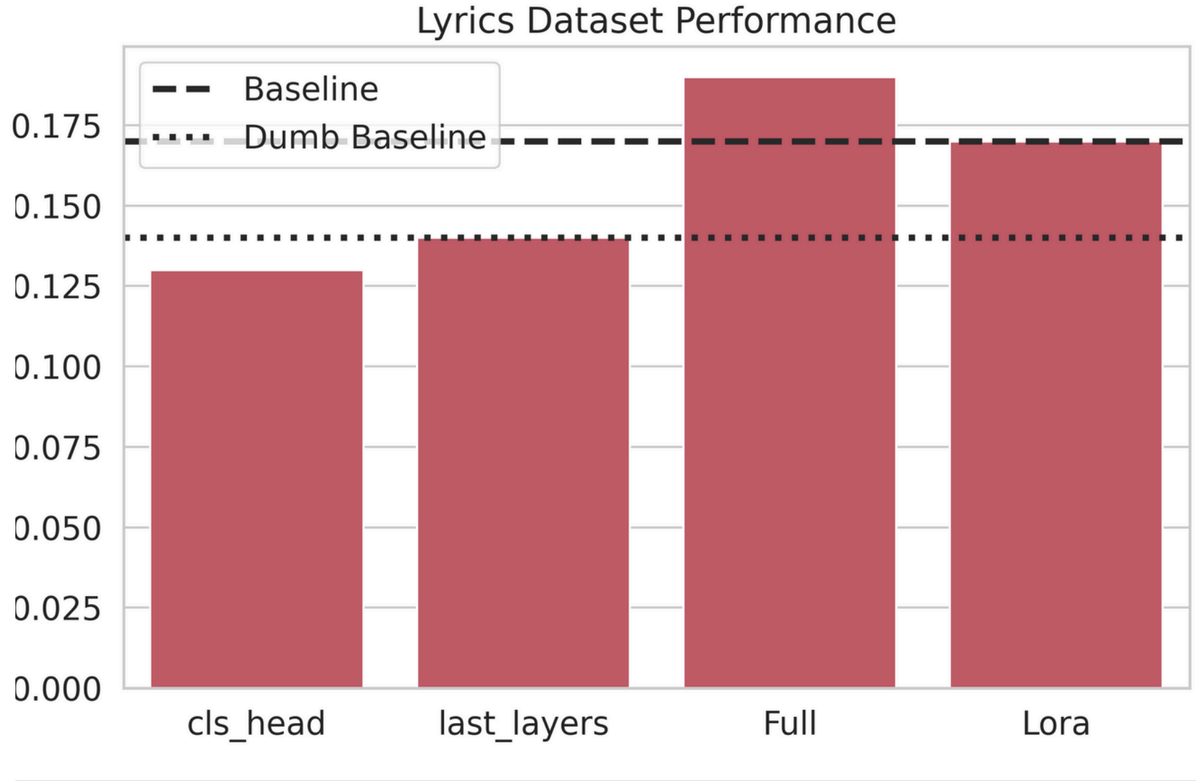
LORA

train a small portion (0.7%) of parameters that matters



RESULTS


Poem Dataset Performance
Baseline
Dumb Baseline
Weighted Avg F1-score
0.0
0.1
0.2
0.3
0.4
0.5
0.6
cls_head
last_layers
Full
Lora


Lyrics Dataset Performance
Baseline
Dumb Baseline
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
cls_head
last_layers
Full
Lora

ratio	f1_perc	f1_songs
0.0	0.611127	0.161053
0.2	0.630270	0.091042
0.4	0.615815	0.158857
0.6	0.590975	0.182402
0.8	0.606499	0.138462
1.0	0.629617	0.123077

ANALYSIS

1. Dataset Extreme Class Imbalance:

- Impact: The model effectively ignores the minority classes.
- Insufficient Data Quantity: 500 samples. For a transformer model like RoBERTa, this is statistically insignificant for robust evaluation. This explains the "spiky" nature of your Validation F1 Score graphs.

2. Model Performance Comparison

- Cls-head only is marginally better than Last N, likely because the pre-trained weights were already robust, and fine-tuning the last layers on such small data introduced noise.
- Despite training only a small portion of parameters, LoRA nearly matches full fine-tuning.
- Training every parameter yielded the best separation of classes, specifically improving the distinction between Joy and Love.

3. Error Analysis

- The "Joy vs. Love" Conflict: Across all models, there is persistent confusion between Joy and Love. Possible reason: Semantically, "Joy" and "Love" share very similar positive sentiment features.
- Training Stability: The Loss Curves (Training Loss) generally look healthy and show convergence for all models. However, the Validation F1 graphs are highly volatile. This suggests that the model is overfitting to the specific small validation batch rather than learning generalizable patterns.

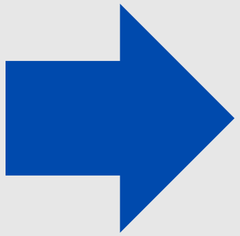
FUTURE WORK

1) Break into different parts
   - Classify separately (chorus, verse, bridge)
   - Voting or average mechanism to produce the final results

```
party_verse = '''I hopped off the plane at LAX
With a dream and my cardigan
Welcome to the land of fame excess (whoa)
Am I gonna fit in?
Jumped in the cab, here I am for the first time
Look to my right, and I see the Hollywood sign
This is all so crazy
Everybody seems so famous'''

party_prechorus = '''My tummy's turning and I'm feeling kinda homesick
Too much pressure and I'm nervous
That's when the taxi man turned on the radio
And a Jay-Z song was on
And a Jay-Z song was on
And a Jay-Z song was on'''

party_chorus = '''So, I put my hands up
They're playing my song, the butterflies fly away
I'm nodding my head like, yeah
Moving my hips like, yeah
I got my hands up, they're playing my song
They know I'm gonna be okay
Yeah, it's a party in the U.S.A.
Yeah, it's a party in the U.S.A.'''
```



```
get_emotion(party_full)
'<pad> sadness'

get_emotion(party_verse)
'<pad> joy'

get_emotion(party_prechorus)
'<pad> sadness'

get_emotion(party_chorus)
'<pad> joy'
```

2) Model handle figurative language, imagery, and metaphors

CONCLUSION

Real World Problem:
Turns raw text (lyrics/poems) into meaningful emotional signals that improve recommendations, wellness tools, creative insights, and NLP performance.

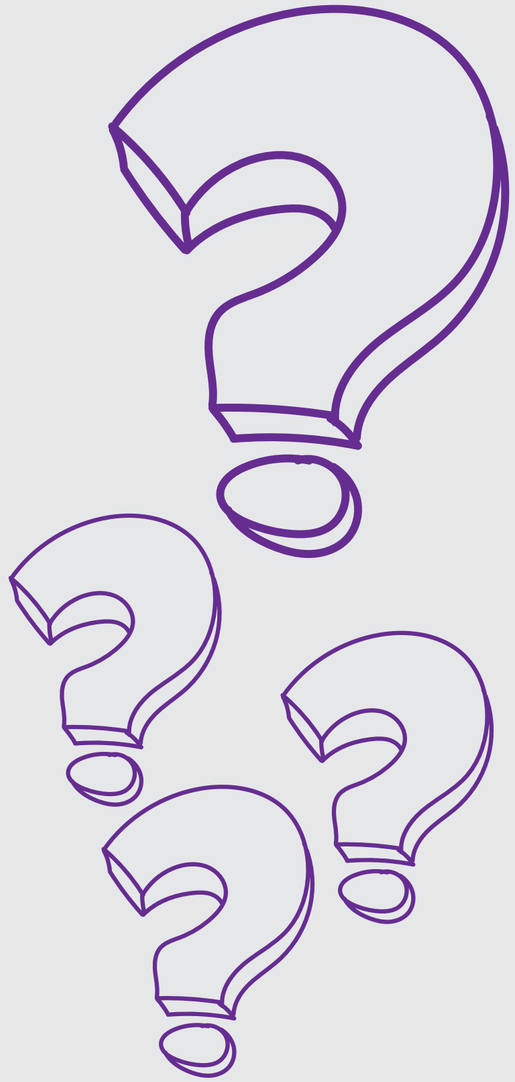
Performance:
Achieved noticeable performance boost (with small dataset)

In depth analysis:
Thorough experiment over different training method
Broke down possible cause of performance limits

Attempted to transfer learning:
Failed but proved a wrong way

Dataset requirements:
Good datasets make a difference in performance

Q&A



Appendix