Multi-Label Emotion Recognition from Text Report:

○ **Dataset preprocessing steps.**

For data preprocessing steps, I have done most of the primary steps to check for outliers, unusual values and even missing values. Then I split the dataset into train, test and validation, after selecting all the emotion columns and making a single list for computing class weights.

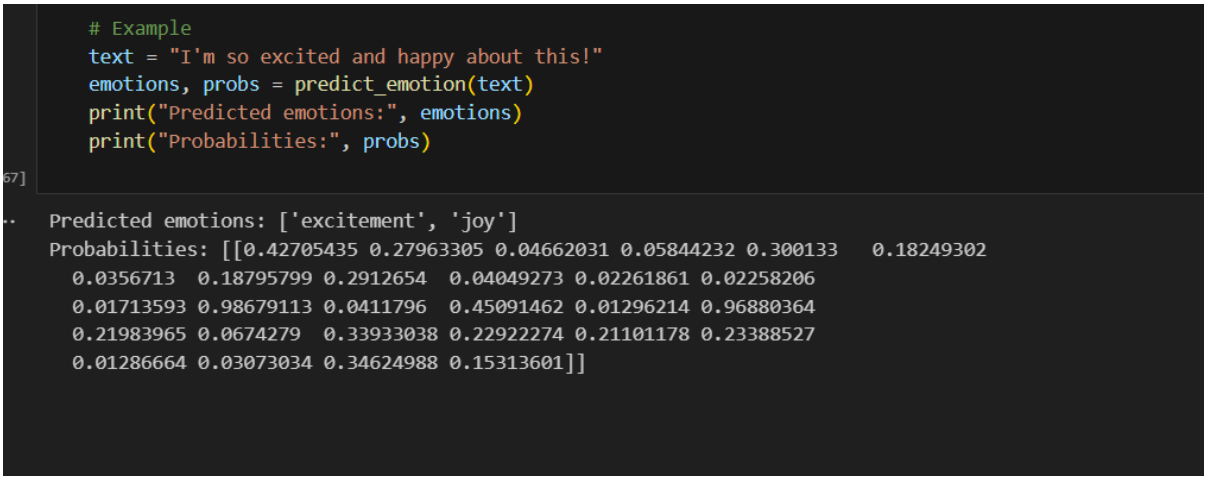
○ **Model selection and rationale.**

For tokenizer, I have used Bert_Tokenizer. And for Model I have used the Bert_For_Sequence_Classification. First, I convert the text strings into list of tokens for easily giving values to the model and training on it. Trained the model on 3 Epochs and learning rate of 2e-5.

○ **Challenges faced and solutions.**

Training the Bert model is quite challenging, as you have to convert the normal dataset to hugging face dataset, then tokenizing it. Then train it, using various hyperparameters and with different values. Was facing difficulty in this step.

○ **Results with visualizations and interpretations.**



```
    # Example
    text = "I'm so excited and happy about this!"
    emotions, probs = predict_emotion(text)
    print("Predicted emotions:", emotions)
    print("Probabilities:", probs)
```

```
Predicted emotions: ['excitement', 'joy']
Probabilities: [[0.42705435 0.27963305 0.04662031 0.05844232 0.300133   0.18249302
  0.0356713  0.18795799 0.2912654  0.04049273 0.02261861 0.02258206
  0.01713593 0.98679113 0.0411796  0.45091462 0.01296214 0.96880364
  0.21983965 0.0674279  0.33933038 0.22922274 0.21101178 0.23388527
  0.01286664 0.03073034 0.34624988 0.15313601]]
```