Text Emotion Classification

By: Valentina Gonzalez Bohorquez & Varun Krishnam

Table of Contents

INTRODUCTION

State the purpose of the project.

- Emotions play vital roles in human existence, as they reflect our current state and well-being.
- There is a need to identify the different emotions expressed by people and use that as the basis to provide recommendations to meet the individual needs of their customers, and ensure business growth.

The purpose of this project is to create an emotion detection system used to automatically recognize emotions in text.

- **Multinomial classification**
- **We used Naive Bayes and a Neural Network as baselines.**
- **BERT was the machine learning architecture used.**

DATA COLLECTION

Discuss the dataset.



- The dataset is publicly available on Kaggle. The authors of the dataset constructed a set of hashtags to collect English tweets from the Twitter API belonging to six emotions: anger, surprise, joy, love, fear, and sadness.
- The dataset consists of about 20,000 tweets, and their corresponding labels.
- The dataset was already cleaned and split into train (16,000 tweets), test (2,000 tweets), and validation set (2,000 tweets).



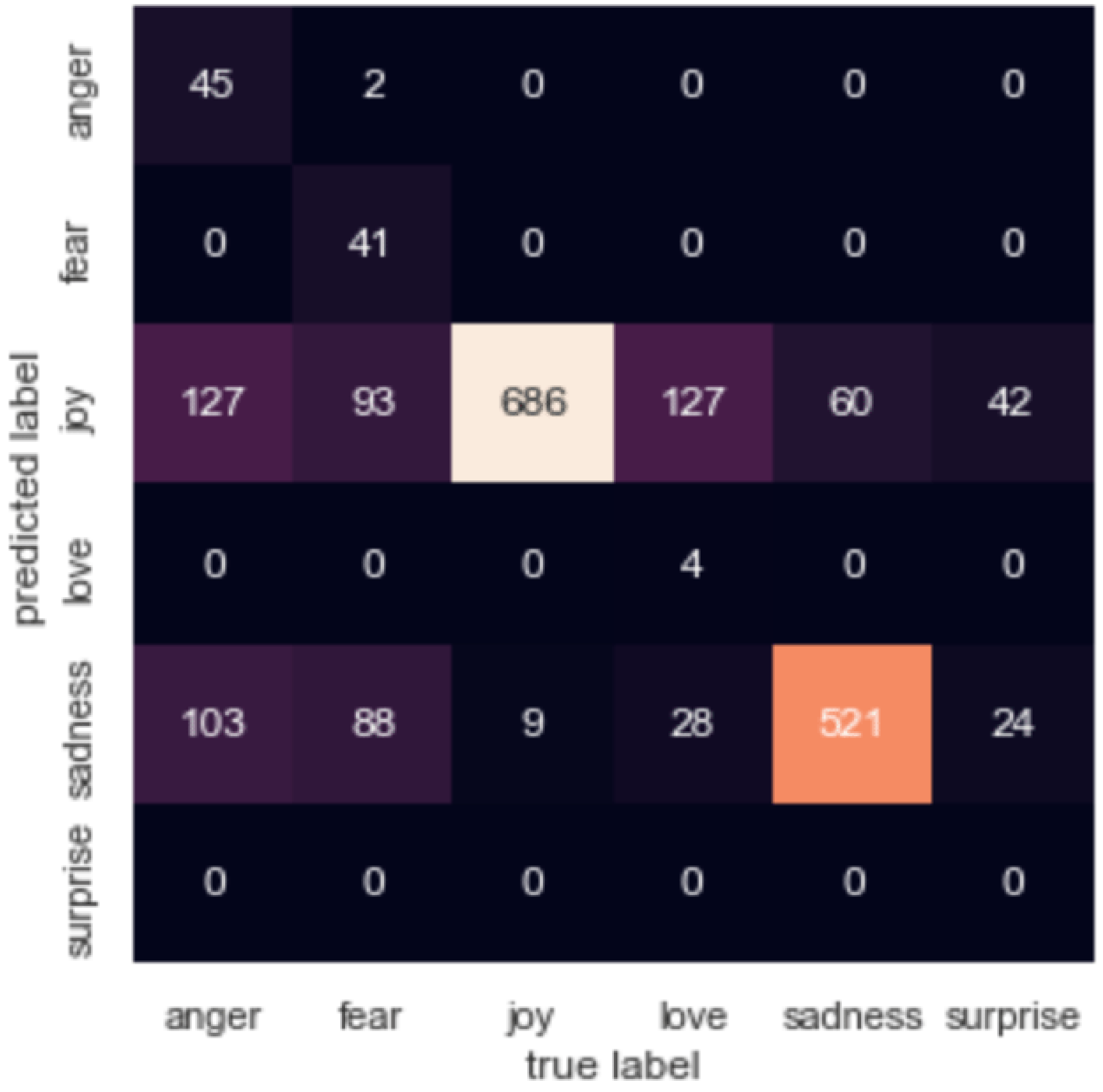
DATA COLLECTION

- Between the training and test data, the emotion distribution is as follows:

joy	6057
sadness	5247
anger	2434
fear	2161
love	1463
surprise	638

NAIVE-BAYES MODEL

- Did well predicting joy & sadness
- Struggled with predicting the other emotions
- Seems to be positive relationship between prediction accuracy and sample data amount

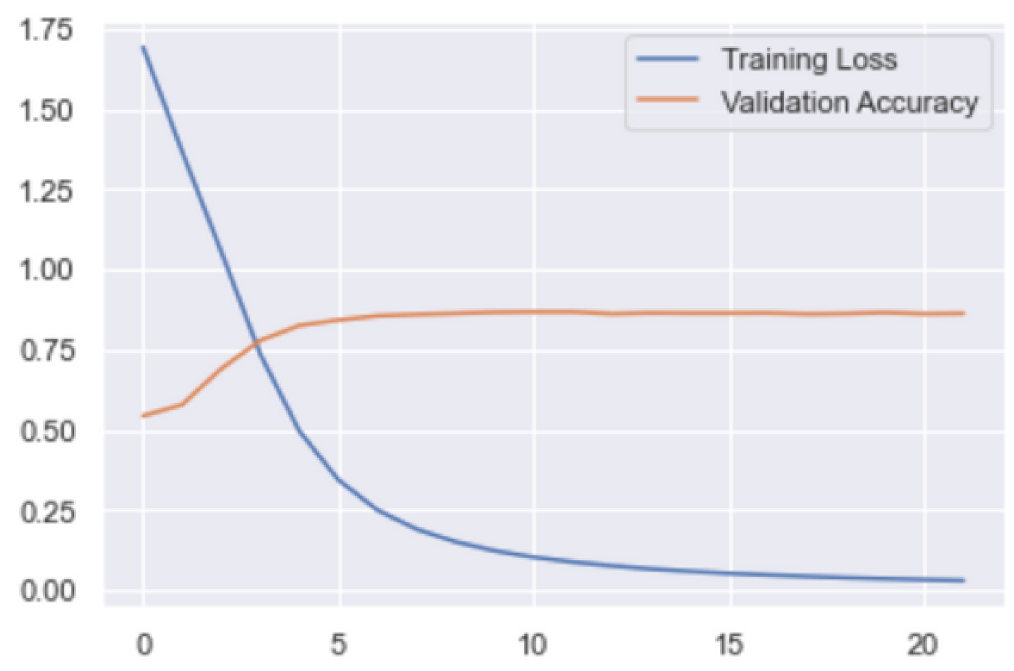


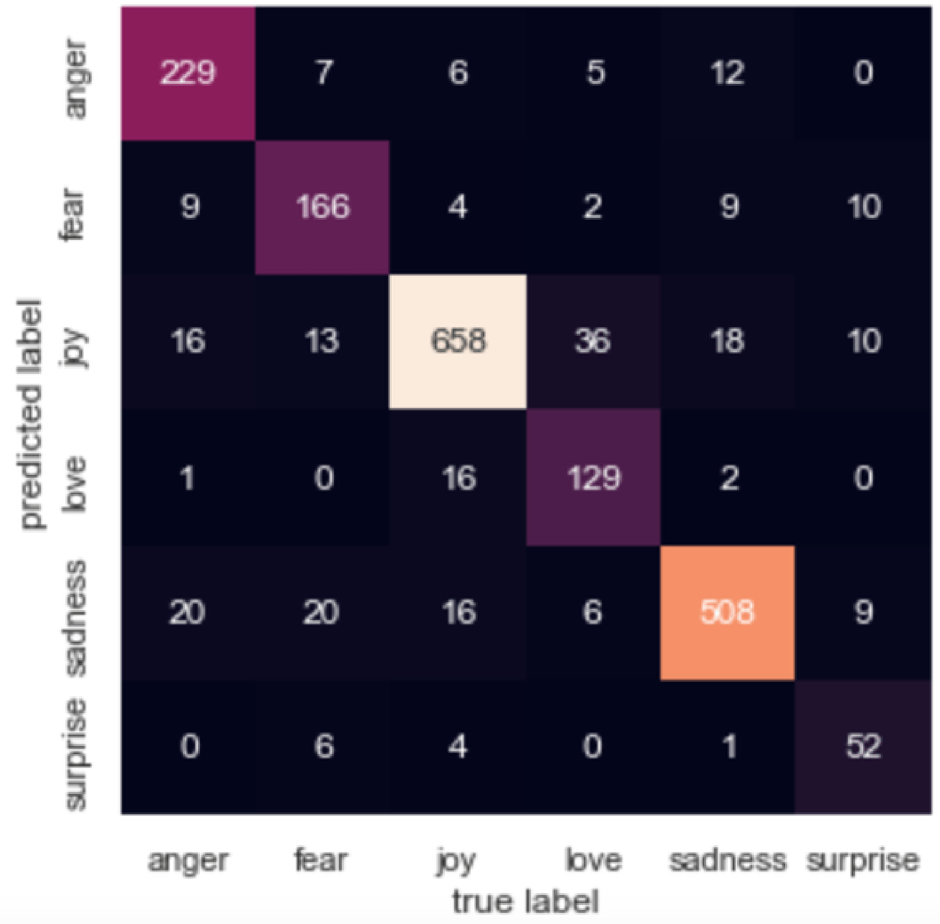
joy	6057
sadness	5247
anger	2434
fear	2161
love	1463
surprise	638

- Micro-averaged F1 score: .6485
- Macro-averaged F1 score: .3595

NEURAL NETWORK MODEL

- Started with a basic architecture
  - 1 hidden layer, 50 units
  - relu activation function
  - training accuracy: 0.977
  - validation accuracy: 0.871





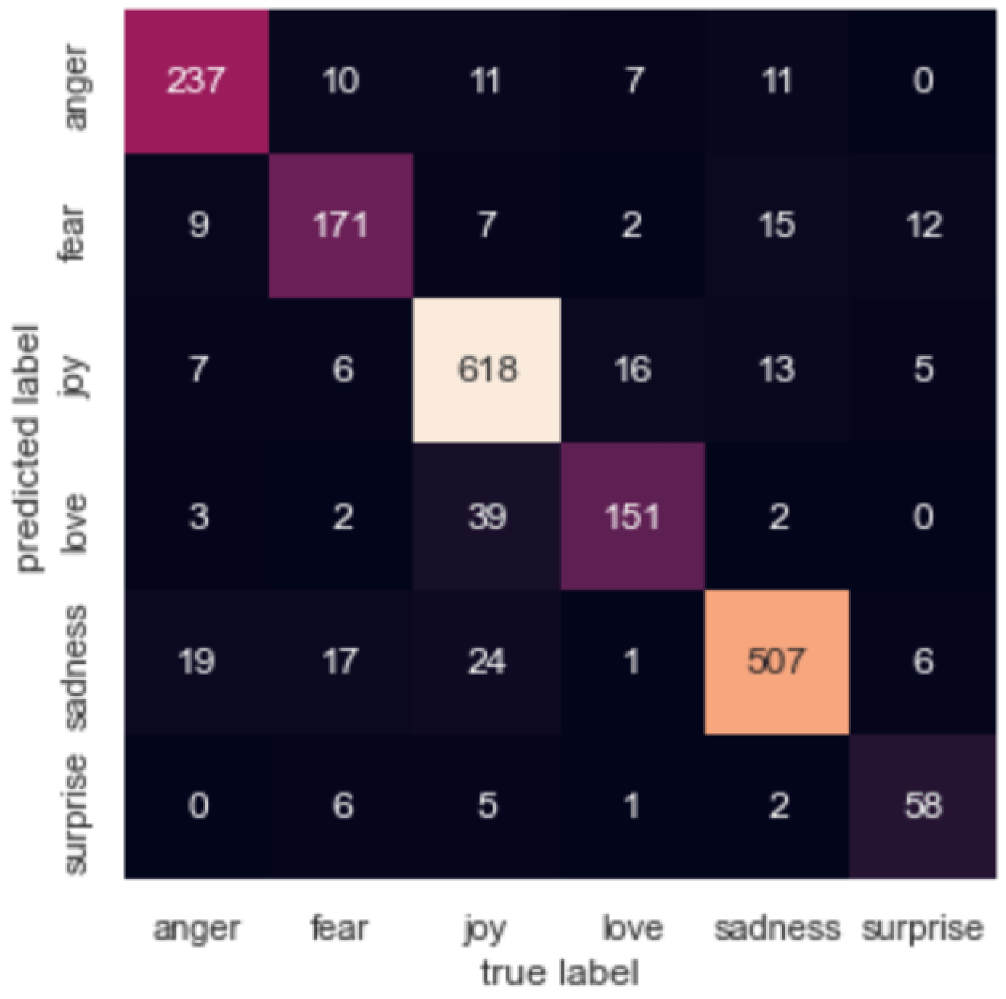
- Micro-averaged F1 score: 0.871
- Macro-averaged F1 score: 0.83

- Training error ~2%
- Validation error ~13%
  - High variance
  - Possible overfitting

- New architecture
- Regularization

UPGRADED NEURAL NETWORK MODEL

- Upgraded architecture
  - 4 hidden layers, 100 units
  - relu activation function
  - Regularization
  - training accuracy: 0.984
  - validation accuracy: 0.87

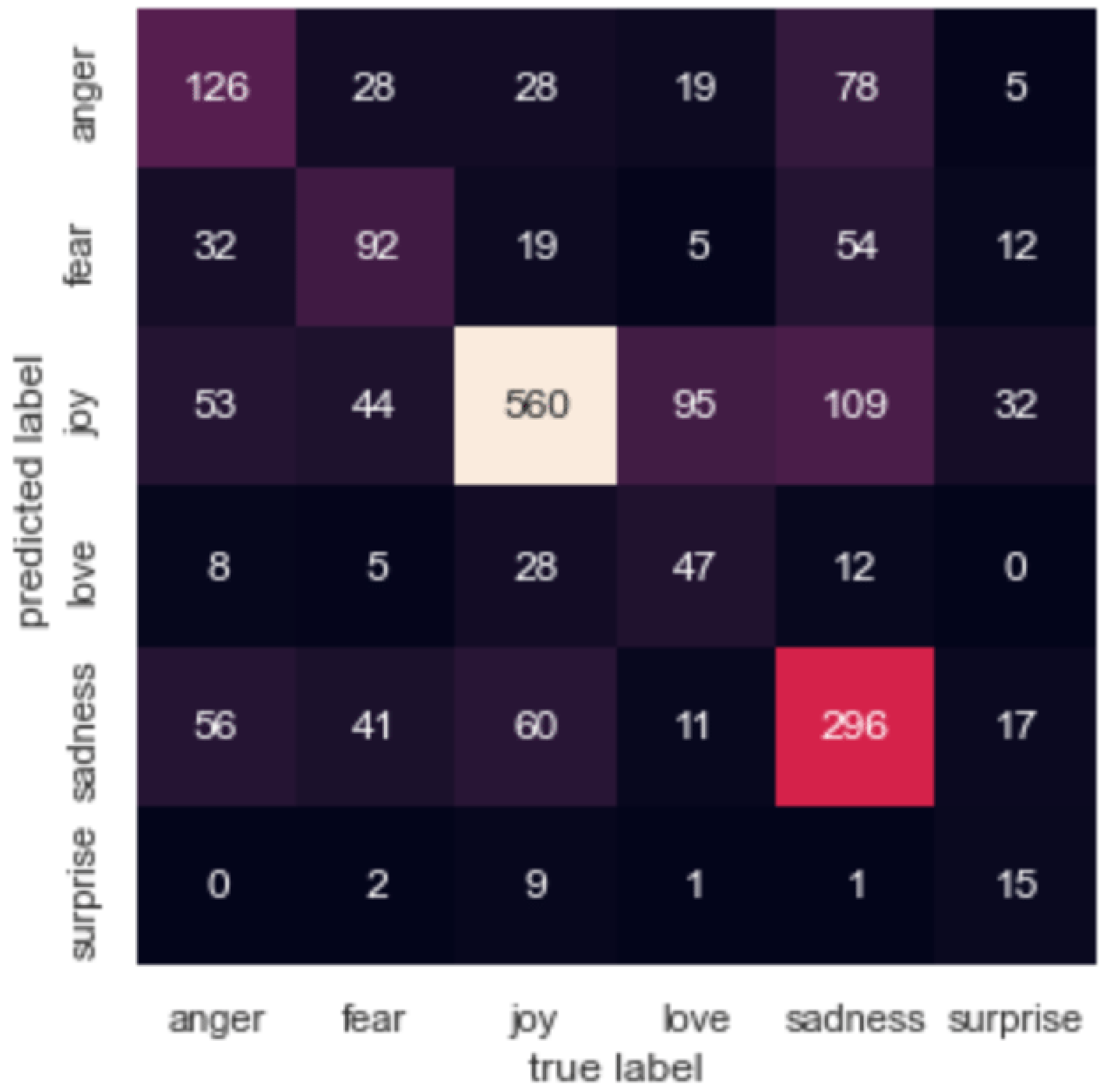


- Micro-averaged F1 score: 0.87
- Macro-averaged F1 score: 0.837

- Very little change in results
- Try new architecture w/ Glove word embeddings & random vectors for OOV words

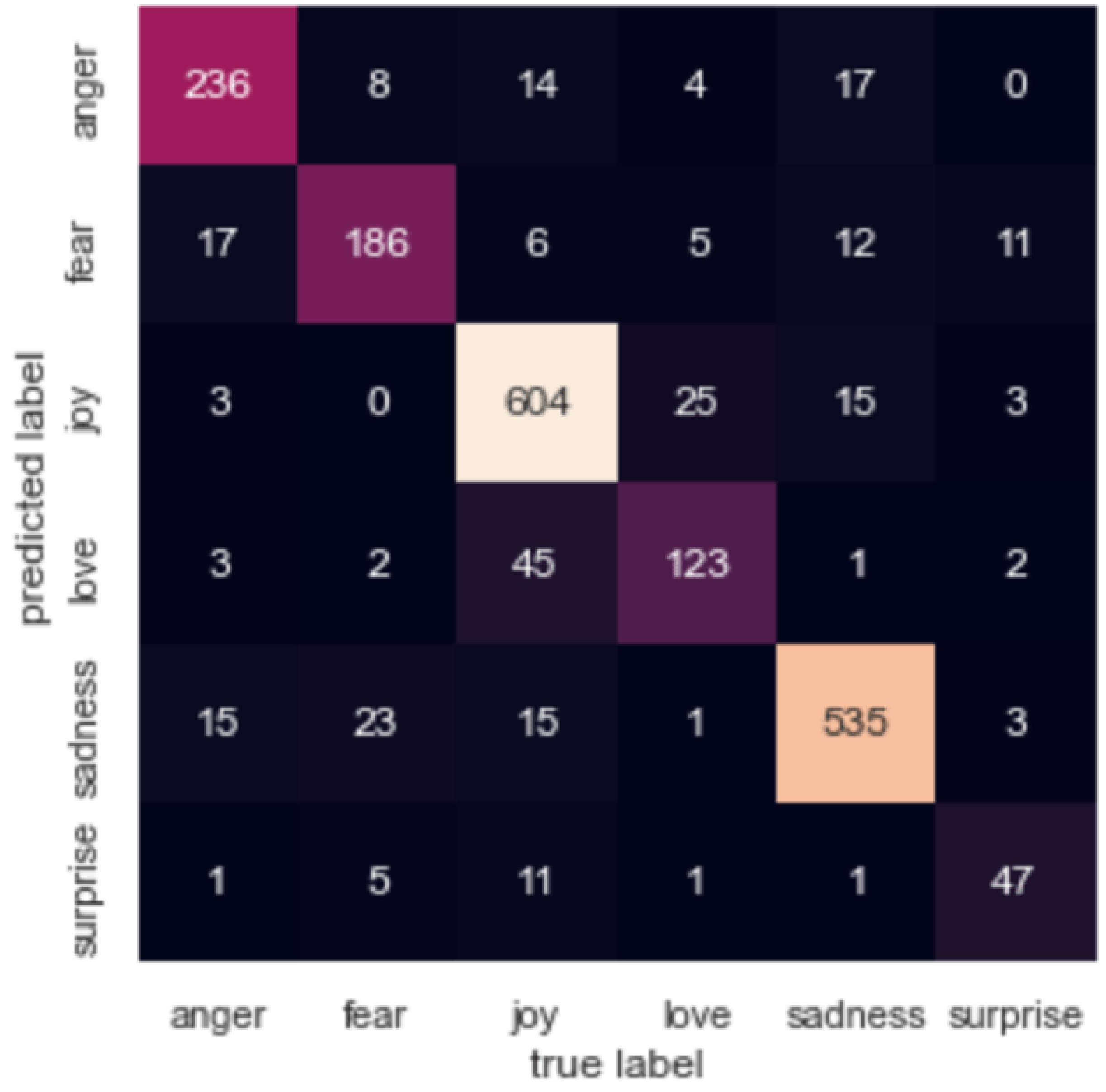
NEURAL NETWORK WITH GLOVE

- Upgraded architecture
  - 4 hidden layers, 600,400,100,50 units
  - relu activation function
  - Regularization
  - Glove word embeddings
  - Random vectors for OOV words
  - training accuracy: 0.624
  - validation accuracy: 0.568



NEURAL NETWORK TEST SET PERFORMANCE

- Micro-averaged F1 score: 0.8655
- Macro-averaged F1 score: 0.819



BERT Model Pre-processing

- Created a dataframe of the already split train and test data sets.
- Modified the unique categorical emotion label values to numerical values

	label	label_encoder
0	sadness	4
2	anger	0
3	love	3
6	surprise	5
7	fear	1
8	joy	2

sentence	label	label_encoder
i didnt feel humiliated	sadness	4
i can go from feeling so hopeless to so damned...	sadness	4
im grabbing a minute to post i feel greedy wrong	anger	0
i am ever feeling nostalgic about the fireplac...	love	3
i am feeling grouchy	anger	0

BERT Model Results

- BERT model performed at 70% accuracy.
- Had to reduce the the training/test dataset to smaller subsets due to long training time.
- We believe BERT model can perform much better if we tune and rerun the model on the entire dataset for the final report.

	precision	recall	f1-score	support
anger	0.64	0.53	0.58	17
fear	0.54	0.58	0.56	12
joy	0.82	0.90	0.86	31
love	0.33	0.20	0.25	5
sadness	0.69	0.75	0.72	32
surprise	1.00	0.33	0.50	3
accuracy			0.70	100
macro avg	0.67	0.55	0.58	100
weighted avg	0.70	0.70	0.69	100

```
array([[ 9,  0,  1,  0,  7,  0],
       [ 1,  7,  1,  0,  3,  0],
       [ 0,  2, 28,  0,  1,  0],
       [ 1,  0,  3,  1,  0,  0],
       [ 3,  2,  1,  2, 24,  0],
       [ 0,  2,  0,  0,  0,  1]])
```

CONCLUSION

- Between the models we have created thus far, the upgraded Neural Network performed the best with a 86.5% accuracy.
- The model that performed the worst was the Neural Network with Glove (56.8% accuracy).
- We expect the BERT model to perform the best when we tune and run the model on the entire train/test dataset.
- In order to improve our results, there needs to be more training data.
  - Balanced classes
  - Large disparity in train data size per emotion