COMM061 – Natural Language Processing

Individual Coursework report

Submitted by:

Venkatesh Nagasubramanian

6721597

vn00197@surrey.ac.uk

Group 3

1. Introduction

This report contains the details and explanations of the code inside the Natural Language Processing coursework. The .zip file submitted along with this sheet contains:

i. 6 Jupyter notebook (.ipynb) files.
ii. 2 Python (.py) files.
iii. 1 helper file (.bin)
iv. 1 file named requirements.txt

1.1. File details

The requirements.txt file contains all the necessary python libraries to run all the code. To install all the libraries, please open a terminal and execute the command:

```
$ pip install -r requirements.txt
```

Once it is installed, you can start running the notebook titled "NLP Coursework – 6721597.ipynb" to view all the outputs.

The five Jupyter notebooks are numbered to denote the order to run the notebooks. The first notebook contains the data analysis and visualization of the dataset. The labels are merged according to the specifics mentioned in the group declaration submitted previously. The labelling guide is, however, mentioned in this report.

The rest of the notebooks contain experiments from 1 through 4. These notebooks will make use of the .py files to run. More about the experiments will be discussed later.

The two .py files have the code to easily implement vectorization and preprocessing techniques. The code is modularized to avoid repetition.

1.2 Label guide

There are 28 labels in the dataset representing 27 emotions and a neutral label. As per the coursework specifications, we were asked to merge the labels into 14 categories. Table 1 summarizes the merging. We have kept the most frequent 14 labels and merged the others amongst the chosen. More about this will be discussed in the Data Analysis and Visualization section.

The labels chosen are Admiration, Amusement, Anger, Annoyance, Approval, Curiosity, Disappointment, Disapproval, Gratitude, Joy, Love, Optimism, Realization, Neutral. The other labels are merged with these with the nearest emotion that they represent from these categories. For example, caring is merged with love.

Now that the labels are clear, we shall move on towards understanding the data.

Emotion Category	Label	Merged emotions	Merged Labels
Admiration	0		
Amusement	1	Surprise	26
Anger	2		
Annoyance	3	Disgust, grief	11, 16
Approval	4	Pride	21
Curiosity	7	Confusion	6
Disappointment	9	Embarrassment, Sadness	12, 25
Disapproval	10	Fear, Nervousness	14, 19
Gratitude	15	Relief	23
Joy	17	Desire, Excitement	8, 13
Love	18	Caring	5
Optimism	20		
Realization	22	Remorse	24
Neutral	27		

Table 1. Labelling guide

2. Data Analysis

The dictionary defines the word 'analysis' as 'detailed examination of the elements or structure of something'. Before we use the data to make predictions, we must understand the data fully so we can make the best decisions and meaningful inferences.

The dataset specified is the GoEmotions dataset [1], which contains 54263 comments in the English language, written by users in reddit. The dataset contains 27 emotions and a neutral label. Each row of text is classified to one or more labels. The frequency of the number of labels that each text is classified is as below:

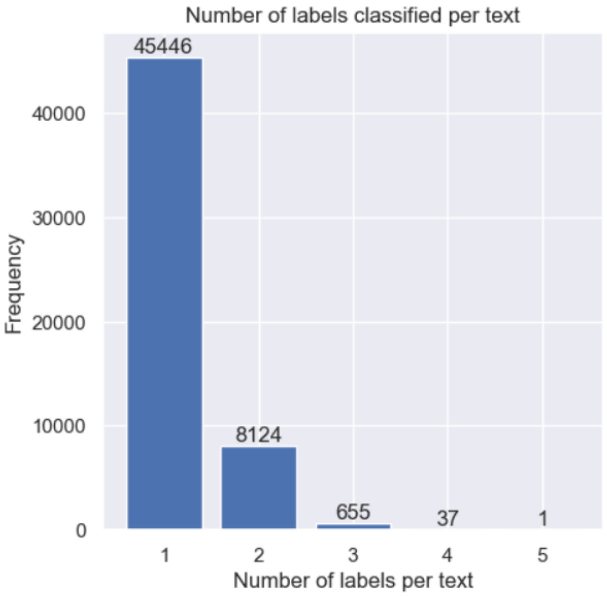


Figure 1. Number of labels classified per comment.

Now, how are the labels distributed? There are two answers to this question. First, in the actual dataset that was downloaded, and second, the distribution of the labels after they were merged.

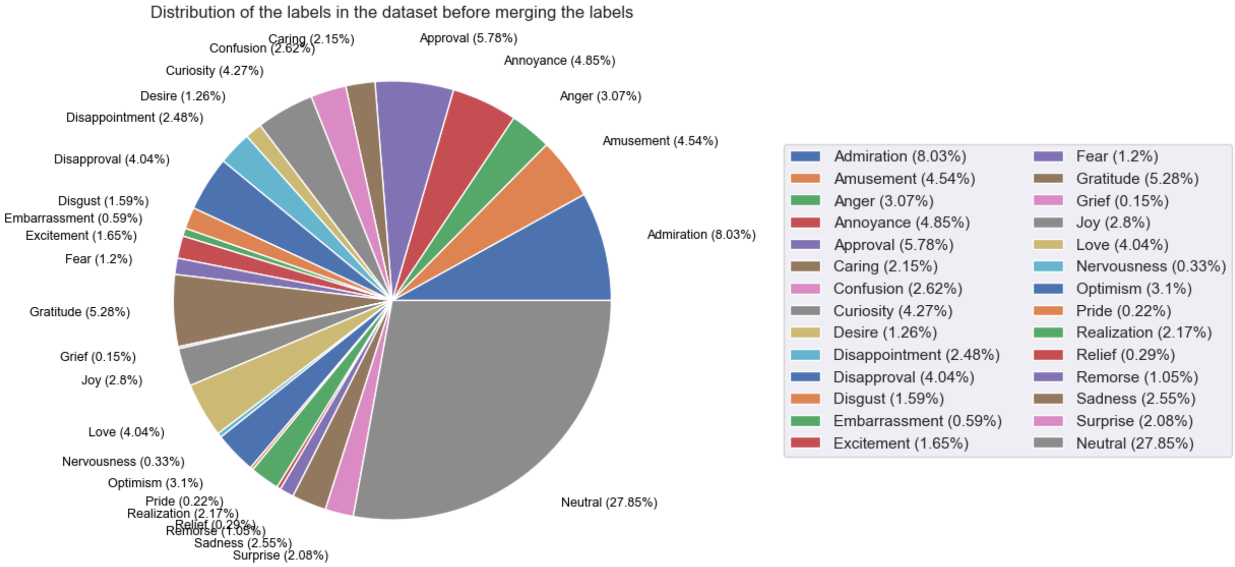


Figure 2. Distribution of the labels in the dataset before merging the labels

The neutral label has 17,772 instances, constituting 27.85% of all the labels with the other labels hardly reaching a third of it. Second in this list, Admiration constitutes 8.05% with 5122 instances. Grief has the lowest number of instances (96) constituting just 0.15% of all the instances.

2.1. Merging the labels

By this time, we know that the labels are merged according to the labelling guide in Table 1. The labels are replaced with the greater category as per the guide. For example, all instances of caring (5) are replaced with Love (18).

As per Figure 1, some of the texts have more than one label. In that case, those labels are reduced to a single label. This is done by selecting the label with the maximum frequency. For example, if a label was, before the merging, [Optimism, Joy, Desire, Neutral], after merging it becomes [Optimism, Joy, Joy, Neutral]. As the most frequent emotion here is Joy, it is considered amongst everything else. This way, each text will be classified into only one label. For 54263 rows, there are 54263 labels.

After merging, the dataset has reduced to 14 labels. The data is now distributed as shown in Figure 3. We can now see that the Neutral label has become much more dominant (29.52%) even though the count has decreased. The emotion with the least number of texts is optimism, with 2.57%. All the labels have now increased in their representation overall.

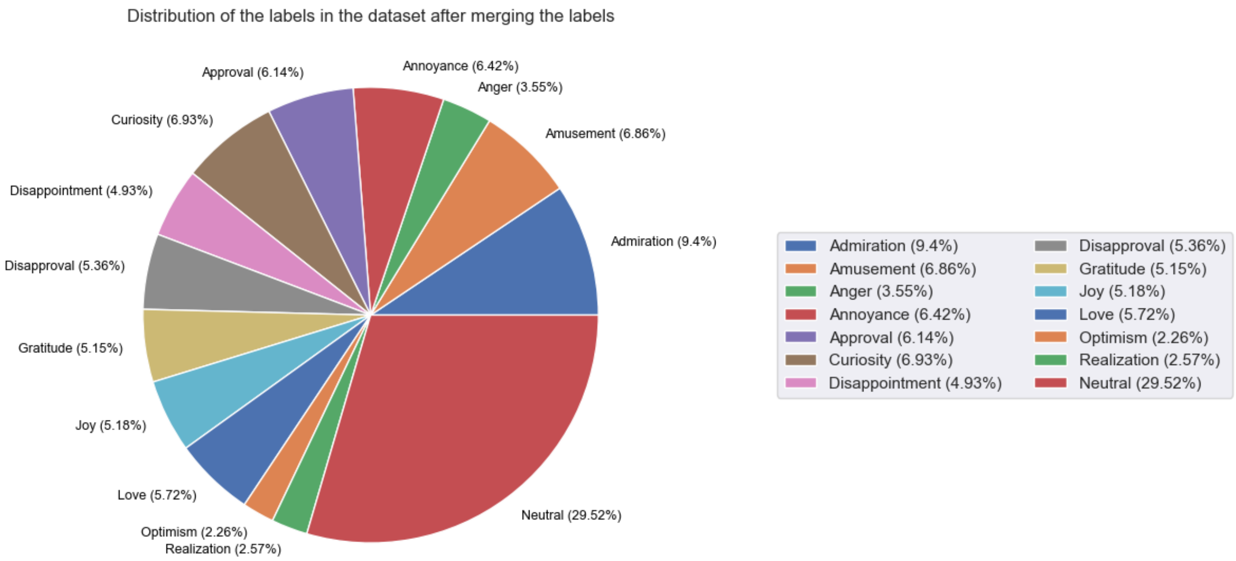


Figure 3. Distribution of the labels in the dataset after merging the labels

2.2 Words

We talked a lot about labels so far. It is time for the most important part of this dataset. Yes, the texts. We already know that these texts are curated from reddit comments. Let us visualize the features of these texts.

One of the best ways to represent texts is to create a word cloud. A word cloud will arrange the words in a text according to the frequency of the words. The more a word appears, the bigger in size it is, relatively. Figure 4 represents the word cloud generated after concatenating all the text from the dataset.

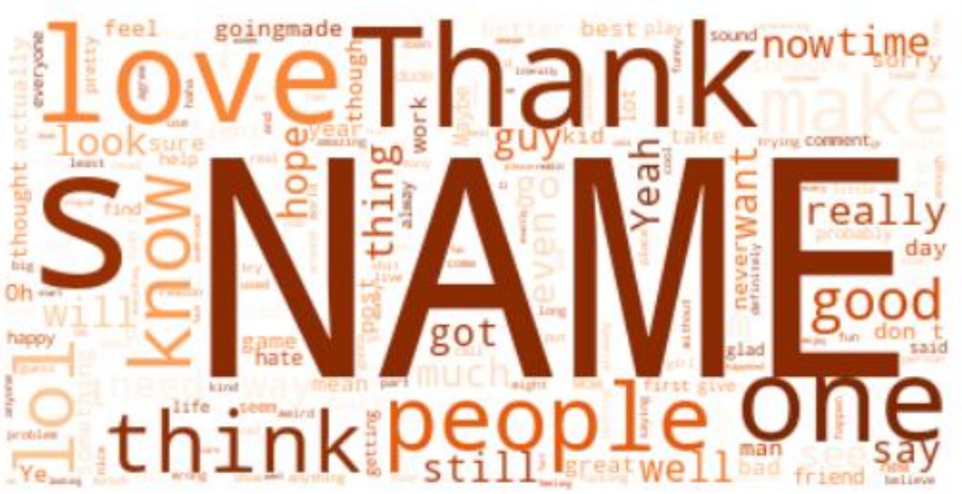
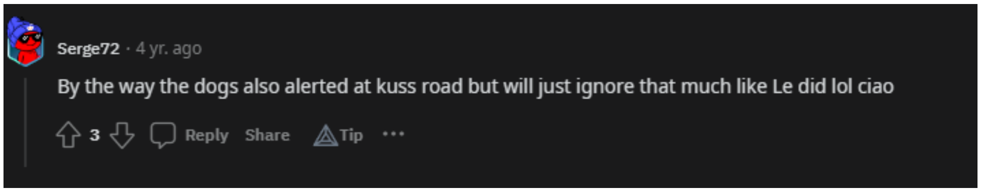


Figure 4. Word cloud of the text data

It seems like the word 'NAME' has an unusual frequency, with 9438 occurrences. One of the texts with the 'NAME' in it, reads like this –

```
By the way the dogs also alerted at kuss road but will just ignore that mu
ch like [NAME] did lol ciao
```

A quick search in Reddit might say that the word ‘NAME’ is a censor for a name of an actual person in that conversation/context. The original comment is like this:



Serge72 · 4 yr. ago

By the way the dogs also alerted at kuss road but will just ignore that much like Le did lol ciao

Figure 5. Original comment by r/Serge72

The word ‘NAME’ was Le, a name from the context of the thread. As this is just a replacement of an actual name, this is heavily repeated and can be used as a stop word. Similarly, other ‘words’ like the word (letter) ‘s’ was also occurring 2840 times. This is also a stop word. Another additional stop word that is replaced from the actual comment is ‘RELIGION’. Both will be removed as a part of stop word removal.

One thing that felt ironic to me was that the word ‘love’ was the fourth highest occurring word, but the label Love is the third least occurring label.

The dataset has been modified and outputted as a .csv file to be used for the experiments.

3. Experiments

In this coursework, there are four experiments that are performed. Each of the experiments has several variations, with the baseline models being the same. Here’s a table to summarize all the baseline models used and the variations.

#	Experiment	Baseline model	Variations
1	Data Pre-processing	tokenization with lemmatization and removal of stop words	1. tokenization with stemming and stop word removal 2. tokenization with stemming, lemmatization and stop word removal 3. tokenization with stemming and lemmatization
2	Training Algorithm	Bi-LSTM	GaussianNB, XGBoost, SVM
3	Text Vectorization	Word2Vec	GloVe
4	Dataset Splitting	75/25	67/33, 50/50, 90/10

Table 2. Baseline models of all experiments with the variations

For every experiment, we shall go through each variation, and compare the models with the baseline variant. We will also delve into some critical analysis of why a certain choice was made for the baseline and the variations. The confusion matrices of the variations are kept at the appendix.

3.1. Baseline models

For every experiment, there is a baseline model, which is compared to the variations. The baseline specifications are the same, but for every experiment, the same model is kept except for the specification of the experiment.

Let’s review the baseline models and discuss the choices which led us to keep the baseline model specification.

Data Preprocessing – Tokenization with stop word removal and lemmatization
Sentences must be tokenized to convert them to vectors. With the removal of stop words, (including the additional stop words and lemmatization of the words, seems to give a better accuracy in the model. The model has given the highest accuracy in this experiment of about 49% on the test data.

Training Algorithm – Bi-directional LSTM
Long short-term memory has been one of the most popular algorithms for any NLP tasks. The bi-directional LSTM model was chosen over the traditional LSTM model because it adds an additional LSTM layer, from which the information flow is reversed.

Demszky et. al. has implemented a bi-LSTM algorithm in [1] for the same dataset and got an F1 score of 0.53. So, it seems like a wise choice to go with their specifications. 20 epochs were chosen as later epochs were overfitting the model. The plots of the epochs vs loss/validation loss were removed to conserve space.

Word2Vec vectorization
Word2Vec is one of the most popular methods of converting words to numerical vectors. The simple answer for choosing Word2Vec over GloVe is that it was much faster to execute and takes a brings much more accuracy to the data.

75-25 Dataset splitting
The dataset is split such that 75% of the data is used for the training and 25% for the testing. It is a common, rule-of-thumb method to split the dataset this way and has generated some good results.

How has the baseline model performed? Let us look at the results.

Class	***Precision***	***Recall***	***F1-Score***	***Support***
Admiration	0.59	0.6	0.59	1302
Amusement	0.53	0.57	0.55	893
Anger	0.45	0.27	0.34	460
Annoyance	0.33	0.21	0.26	874
Approval	0.3	0.11	0.16	760
Curiosity	0.49	0.08	0.13	957
Disappointment	0.4	0.3	0.34	667
Disapproval	0.5	0.11	0.18	733
Gratitude	0.8	0.79	0.79	712
Joy	0.43	0.37	0.4	660
Love	0.55	0.47	0.51	812
Optimism	0.45	0.36	0.4	337
Realization	0.5	0.16	0.24	327
Neutral	0.46	0.8	0.59	4072
Accuracy			0.49	13566
Macro avg	0.49	0.37	0.39	13566
Weighted avg	0.48	0.49	0.45	13566

Table 3. Classification report of the baseline model

The model has performed very well with 49% accuracy. From this report, we can say that the model is able to identify Admiration, Amusement, and Gratitude with a better accuracy and does a poor job in identifying emotions like Curiosity, Annoyance and Disappointment. Below is the confusion matrix of the same.

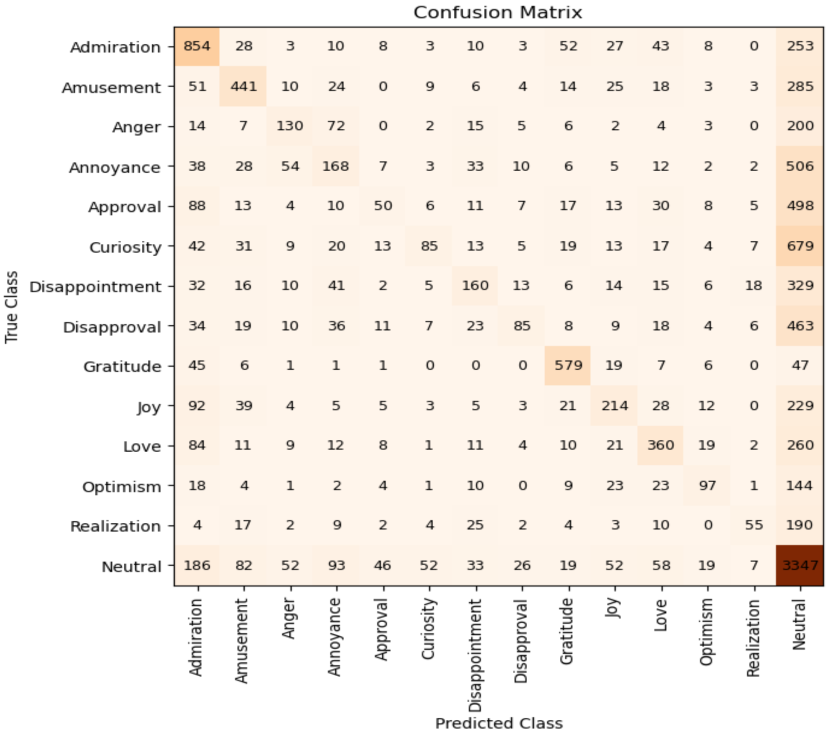


Figure 5. Confusion matrix of the baseline model

3.2. Experiment 1 – Data Preprocessing

Data Preprocessing is the initial part of constructing an NLP model. The texts in the dataset are processed so that it includes only the meaningful information in the texts. For all models, the punctuation is removed, and the texts are tokenized.

All the data preprocessing methods are built into the preprocessing.py file. The file consists of certain functions, like tokenization, stemming, lemmatization and stop word removal. The package used to implement these techniques is called `nltk`. This package consists of all the required operations that can be used to effectively preprocess the models.

Let's now look at all the variations and how they have been performed.

i. Stemming with the removal of additional stop words. Stemming the words will remove the last few letters of the words, like -ing or plurals that end with -s or -es. The upside is that by stemming, words like running become run, which is the root form of the word. The downside, however, is that Moses becomes Mos. Stop words have also been removed, like I, an, the to include only the meaningful parts of the sentence. This model has performed as follows:

Class	***Precision***	***Recall***	***F1-score***	***Support***
Admiration	0.56	0.38	0.46	1302
Amusement	0.58	0.51	0.54	893
Anger	0.49	0.27	0.35	460
Annoyance	0.35	0.15	0.21	874
Approval	0.28	0.07	0.12	760
Curiosity	0.35	0.05	0.09	957
Disappointment	0.42	0.22	0.29	667
Disapproval	0.45	0.03	0.06	733
Gratitude	0.79	0.83	0.81	712
Joy	0.45	0.25	0.33	660
Love	0.54	0.49	0.51	812
Optimism	0.5	0.35	0.41	337
Realization	0	0	0	327
Neutral	0.41	0.85	0.55	4072
Accuracy			0.46	13566
Macro avg	0.44	0.32	0.34	13566
Weighted avg	0.45	0.46	0.40	13566

Table 4. Classification report of the experiment #1, variation #2

ii. Stemming and lemmatization with the removal of additional stopwords. Lemmatization, like stemming will also reduce the words to their root form. Lemmatization takes into consideration the context of the sentence. For example, ‘I am’ can be reduced to the verb ‘be’. The model has performed as follows:

Label	***Precision***	***Recall***	***F1-score***	***Support***
Admiration	0.46	0.44	0.45	1302
Amusement	0.6	0.4	0.48	893
Anger	0.39	0.35	0.37	460
Annoyance	0.37	0.14	0.2	874
Approval	0.27	0.07	0.12	760
Curiosity	0.39	0.05	0.09	957
Disappointment	0.38	0.28	0.32	667
Disapproval	0.4	0.04	0.07	733
Gratitude	0.74	0.85	0.79	712
Joy	0.38	0.26	0.31	660
Love	0.56	0.43	0.49	812
Optimism	0.43	0.39	0.41	337
Realization	0	0	0	327
Neutral	0.42	0.81	0.55	4072
Accuracy			0.45	13566
Macro avg	0.41	0.32	0.33	13566
Weighted avg	0.43	0.45	0.39	13566

Table 5. Classification report of experiment #1, variation #3

iii. Tokenization with stemming and lemmatization. This variation will check the overall usage of the stop words in the dataset. The model has performed as follows:

Label	*Precision*	*Recall*	*F1-score*	*Support*
Admiration	0.53	0.4	0.45	1302
Amusement	0.59	0.48	0.53	893
Anger	0.41	0.33	0.37	460
Annoyance	0.38	0.13	0.2	874
Approval	0.28	0.08	0.12	760
Curiosity	0.38	0.04	0.08	957
Disappointment	0.45	0.18	0.26	667
Disapproval	0.46	0.05	0.1	733
Gratitude	0.79	0.84	0.81	712
Joy	0.44	0.27	0.33	660
Love	0.57	0.44	0.5	812
Optimism	0.48	0.37	0.42	337
Realization	0.14	0	0.01	327
Neutral	0.41	0.85	0.56	4072
Accuracy	-	-	0.46	13566
Macro avg	0.45	0.32	0.34	13566
Weighted avg	0.45	0.46	0.4	13566

Table 6. Classification report of experiment #1, variation #4

Discussion of Results

What do we infer from all these tables? We can see that classes like Realization and Curiosity have been the least predicted classes. The f1 scores of these classes have been very low, almost close to zero. The models have not been able to understand the patterns with these classes. However, classes like Gratitude were mostly predicted correctly. The baseline model has done better in terms of accuracy than the others. Below is a bar chart representation of the accuracies.

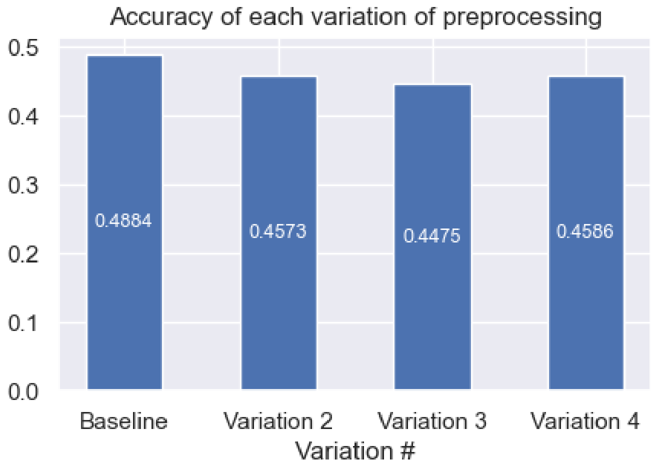


Figure 6. Accuracy of each variation of preprocessing

3.3. Experiment 2 – Training Algorithms

One of the most important parts of building a model is training and choosing the best algorithm. Here, the baseline model is the bi-LSTM, the results of which are shared in Table 3.

i. Tree-based models have been one of the popular choices for NLP models. *Olah et. al.* implemented an XGBoost algorithm in [2] which was one of the baseline models for their comparisons. The results were as follows:

Label	**Precision**	**Recall**	**F1-score**	**Support**
Admiration	0.58	0.52	0.55	1302
Amusement	0.56	0.4	0.47	893
Anger	0.58	0.18	0.28	460
Annoyance	0.31	0.11	0.17	874
Approval	0.25	0.05	0.08	760
Curiosity	0.47	0.06	0.1	957
Disappointment	0.41	0.14	0.21	667
Disapproval	0.45	0.05	0.09	733
Gratitude	0.79	0.71	0.75	712
Joy	0.52	0.23	0.32	660
Love	0.58	0.36	0.44	812
Optimism	0.5	0.2	0.29	337
Realization	0.51	0.12	0.19	327
Neutral	0.4	0.9	0.56	4072
Accuracy			0.45	13566
Macro avg	0.49	0.29	0.32	13566
Weighted avg	0.47	0.45	0.39	13566

Table 7. Classification report of Experiment #2, Variation #1 (XGBoost)

ii. As another popular algorithm for the NLP tasks, Naïve Bayes was sought out as a variation for this experiment. Here are the results:

Label	**Precision**	**Recall**	**F1-score**	**Support**
Admiration	0.63	0.3	0.41	1302
Amusement	0.26	0.43	0.32	893
Anger	0.15	0.32	0.21	460
Annoyance	0.23	0.17	0.19	874
Approval	0.12	0.18	0.15	760
Curiosity	0.19	0.04	0.06	957
Disappointment	0.26	0.13	0.18	667
Disapproval	0.25	0.05	0.08	733
Gratitude	0.8	0.48	0.6	712
Joy	0.4	0.16	0.23	660
Love	0.55	0.32	0.4	812
Optimism	0.06	0.7	0.11	337
Realization	0.12	0.29	0.17	327
Neutral	0.53	0.26	0.35	4072
Accuracy			0.26	13566
Macro avg	0.32	0.27	0.25	13566
Weighted avg	0.4	0.26	0.28	13566

Table 8. Classification report of Experiment #2, Variation #3 (Naïve Bayes)

iii. SVM classifier was another baseline model that was stated in [2]. In their experiments, SVM gave a very high average recall of 0.61. Hence, this was chosen to be experimented with. Here are the results:

Label	Precision	Recall	F1-score	Support
Admiration	0.58	0.62	0.6	1302
Amusement	0.62	0.53	0.57	893
Anger	0.55	0.25	0.34	460
Annoyance	0.39	0.14	0.21	874
Approval	0.42	0.03	0.06	760
Curiosity	0.63	0.03	0.06	957
Disappointment	0.48	0.19	0.27	667
Disapproval	0.72	0.05	0.08	733
Gratitude	0.8	0.81	0.81	712
Joy	0.53	0.31	0.39	660
Love	0.6	0.45	0.52	812
Optimism	0.53	0.32	0.4	337
Realization	0.55	0.17	0.25	327
Neutral	0.43	0.9	0.58	4072
Accuracy			0.49	13566
Macro avg	0.56	0.34	0.37	13566
Weighted avg	0.53	0.49	0.43	13566

Table 9. Classification report of Experiment #2, Variation #4 (SVM Classifier)

Discussion of results

SVM has performed a little better in terms of accuracy compared to the bi-LSTM model. But the disadvantage, however, is that SVM Classifier took four times the time to run than the bi-LSTM. Due to the time complexity, the SVM was not chosen as the better model.

As seen earlier all the models have struggled with identifying the classes, like Curiosity and Disapproval. In contrast to the baseline model, the SVM has been able to predict realization much better. Once again, Gratitude has become the highest predicted class. Below is the accuracy of all the models in this experiment.

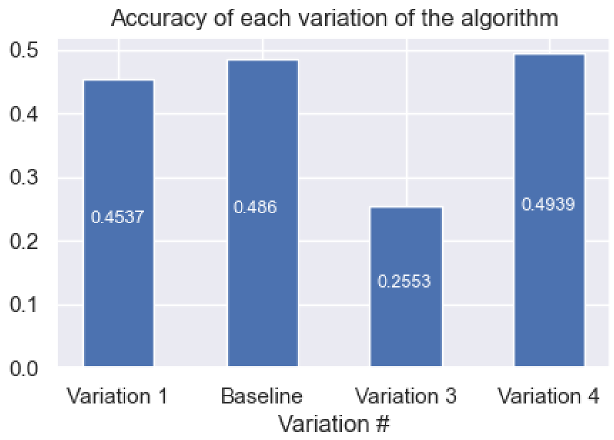


Figure 7. Accuracy of each variation of training algorithm

3.4. Experiment 3 – Text vectorization

Machine learning and deep learning models can only understand and find patterns in numbers. Hence, to classify text data, they must be converted to numbers. And that is the purpose of text vectorization.

The only other variation from the baseline model here is the GloVe. This is vectorized using the 'glove-twitter-200' pre-trained model [3]. The model performed as follows.

Class	Precision	Recall	F1-Score	Support
Admiration	0.58	0.6	0.59	1302
Amusement	0.56	0.55	0.56	893
Anger	0.47	0.3	0.37	460
Annoyance	0.34	0.23	0.28	874
Approval	0.36	0.09	0.14	760
Curiosity	0.44	0.08	0.14	957
Disappointment	0.41	0.25	0.31	667
Disapproval	0.5	0.1	0.17	733
Gratitude	0.77	0.8	0.78	712
Joy	0.46	0.31	0.37	660
Love	0.57	0.45	0.5	812
Optimism	0.46	0.29	0.36	337
Realization	0.51	0.18	0.27	327
Neutral	0.45	0.81	0.58	4072
Accuracy			0.49	13566
Macro avg	0.49	0.36	0.39	13566
Weighted avg	0.48	0.49	0.44	13566

Table 10. Classification report of Experiment #3 Variation #2 (GloVe)

Results and Discussion

Both the models have performed similarly in terms of accuracy. But when you look at the time elapsed for each of the models, GLoVe has taken around six times the time Word2Vec had taken. This is again a question of time complexity, which makes the baseline model the better of the two.

As seen earlier, the GloVe model has performed poorly in the same classes as the Word2Vec or the baseline model. But an important point to note is that the baseline model has a better f1 score than the GloVe model.

Below is a bar chart of the accuracies of the two variations.

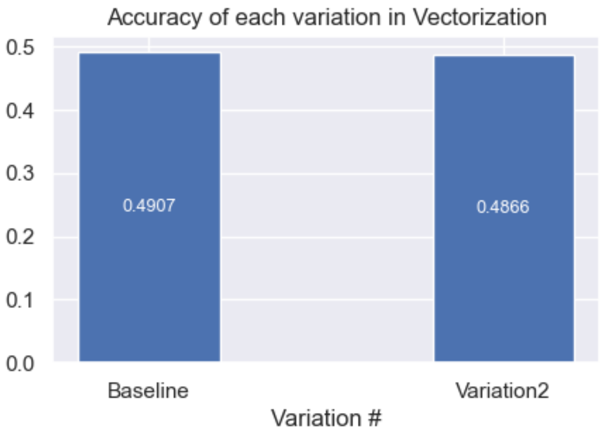


Figure 8. Accuracy of each variation in vectorization

3.5. Experiment 4 – Dataset splitting.

Before feeding the data to the model, it is vital to split the dataset to train and test data. The training data is fed to the model so that it can learn from the model. The test data is kept unseen to the model so we can evaluate the performance over the unseen data.

The variation in each dataset is the percentage of split in the data. The random_state parameter is set to 14 to get the same outputs everywhere. The baseline model was split 75% for the training data and 25% for the test data. Here are the variations and the results:

i. Variation 2: Splitting the data to 67% for training and 33% for test. The results were as follows.

Class	Precision	Recall	F1-Score	Support
Admiration	0.55	0.63	0.59	1701
Amusement	0.59	0.53	0.56	1177
Anger	0.4	0.4	0.4	617
Annoyance	0.35	0.17	0.23	1164
Approval	0.33	0.09	0.15	1070
Curiosity	0.46	0.08	0.14	1271
Disappointment	0.45	0.25	0.32	882
Disapproval	0.52	0.09	0.16	952
Gratitude	0.78	0.76	0.77	913
Joy	0.5	0.31	0.38	890
Love	0.59	0.43	0.5	1059
Optimism	0.52	0.31	0.39	437
Realization	0.46	0.23	0.3	435
Neutral	0.45	0.82	0.58	5339
Accuracy			0.49	17907
Macro Avg	0.5	0.37	0.39	17907
Weighted Avg	0.49	0.49	0.44	17907

Table 11. Classification report of Experiment #4, variation #2 (67-33 split)

ii. Variation #3: Splitting the dataset to 90% train and 10% test data. The results were as follows.

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Admiration	0.55	0.65	0.6	537
Amusement	0.51	0.6	0.55	324
Anger	0.44	0.3	0.36	188
Annoyance	0.35	0.22	0.27	348
Approval	0.37	0.1	0.16	306
Curiosity	0.5	0.07	0.12	355
Disappointment	0.42	0.32	0.36	255
Disapproval	0.44	0.11	0.18	302
Gratitude	0.76	0.81	0.78	288
Joy	0.43	0.33	0.37	270
Love	0.56	0.46	0.51	328
Optimism	0.52	0.43	0.47	143
Realization	0.52	0.23	0.32	145
Neutral	0.48	0.79	0.59	1638
Accuracy			0.5	5427
Macro Avg	0.49	0.39	0.4	5427
Weighted Avg	0.49	0.5	0.46	5427

Table 12. Classification report of Experiment #4, variation #3 (90-10 split)

iii. Variation #4: Splitting the dataset in half (50% train and 50% test). The results were as follows.

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Admiration	0.55	0.63	0.59	2550
Amusement	0.6	0.53	0.56	1863
Anger	0.48	0.31	0.38	972
Annoyance	0.36	0.16	0.22	1747
Approval	0.31	0.09	0.14	1638
Curiosity	0.35	0.11	0.17	1904
Disappointment	0.43	0.27	0.33	1357
Disapproval	0.37	0.12	0.18	1408
Gratitude	0.79	0.74	0.76	1388
Joy	0.52	0.31	0.39	1387
Love	0.53	0.47	0.5	1556
Optimism	0.53	0.32	0.4	641
Realization	0.45	0.17	0.25	656
Neutral	0.44	0.79	0.57	8065
Accuracy			0.48	27132
Macro avg	0.48	0.36	0.39	27132
Weighted avg	0.47	0.48	0.44	27132

Table 13. Classification report of Experiment #4, variation #4 (50-50 split)

Discussion of results

No matter what experiments we do, which variation we choose, the model is very bad in predicting classes like Approval, Curiosity and Disappointment. Once again, Gratitude and Admiration are predicted much more accurately.

Variation #3 has the best accuracy of them all, even surpassing the baseline model. The reason for this could be that there was a lot of data for the model to train and very less data to test which made it easier for the model to predict better. Here is the comparison of accuracy of all the variations in this experiment.

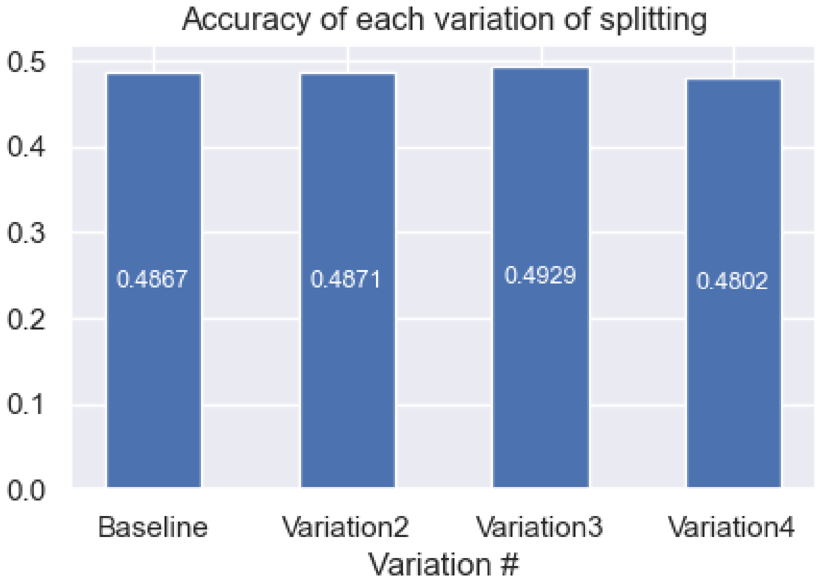


Figure 9. Accuracy of each variation in splitting

4. Conclusion and future work

The baseline model that we have set has given an accuracy of 49%. Although there are other variations that have given slightly better accuracies, it is important to note the time taken to run these algorithms. In experiment 2, it was surprising to see SVC run faster than a neural network model. This could have a simple explanation – the bi-LSTM model uses TensorFlow which is configured with a GPU in the system used to run all these models and SVM might not use the GPU hence it is slower than expected.

System specifications: CPU - Intel i7-12700H (14 cores, 20 threads), GPU – Nvidia RTX 3050 4GB, RAM – 16GB DDR4.

So, which is the best model? The baseline model has the best accuracy of all the models and takes less time to run the model. Experiment #4 Variation #3 has given a slightly better accuracy but feeding much more data to the model also means much more time to run the model.

4.1. Improvements

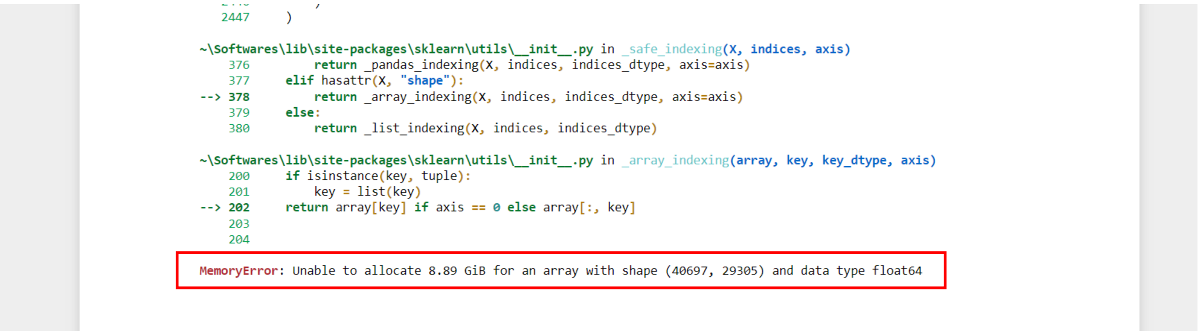
As much an effort that went into making this model, it is still not perfect at all. Not by a long shot. The highest accuracy achieved in all the models is just 49.82%, which is very low. But why did that happen? There are a few reasons to it:

- Initially the dataset had 28 labels, which were reduced to 14. This led the data from other labels to be merged into categories which might not share the text pattern from the original label. This might also be the reason why classes like Approval, Curiosity and Disappointment were incorrectly predicted most of the time. More research is required to optimize the merging of the labels.
- There is a high disparity in the count of the labels. The neutral label takes up around 25-30% of all the data which might confuse the model to predict effectively.

In the variations, you might notice that the baseline accuracies are different. This is due to the dropout layer being added onto the bi-LSTM model. Hence the model fluctuates with accuracy.

4.2. Future work

With all the experiments done, some of the variations that can be done to improve the model's performance are always prevalent. You may have noticed that there are only 2 variations that were there in the vectorization experiment. TF-IDF was initially another variation that was considered for this process, but it was removed later as the vectors had around 29,000 columns and needed much more compute power to run a bi-LSTM model. Figure 10 shows the error message that says that the system is unable to allocate memory for the process.



```
   2447     )

~\Softwares\lib\site-packages\sklearn\utils\__init__.py in _safe_indexing(X, indices, axis)
    376         return _pandas_indexing(X, indices, indices_dtype, axis=axis)
    377     elif hasattr(X, "shape"):
--> 378         return _array_indexing(X, indices, indices_dtype, axis=axis)
    379     else:
    380         return _list_indexing(X, indices, indices_dtype)

~\Softwares\lib\site-packages\sklearn\utils\__init__.py in _array_indexing(array, key, key_dtype, axis)
    200     if isinstance(key, tuple):
    201         key = list(key)
--> 202     return array[key] if axis == 0 else array[:, key]
    203
    204

MemoryError: Unable to allocate 8.89 GiB for an array with shape (40697, 29305) and data type float64
```

Figure 10. Error running TF-IDF

In reddit, writing a comment with a different case might give out a different emotion. For example, the text in the dataset that says,

```
OmG pEyToN iSn'T gOoD eNoUgH tO hElP uS iN tHe PlAyOfFs! Dumbass Broncos
fans circa December 2015.
```

In that sentence, the mixed capitals mean a sarcastic comment or a mockery of someone. Here it is labelled as amusement. While preprocessing the data, both stemming and lemmatization from the nltk library will convert this to lower case. This will remove the meaning of the sentence and doesn't capture the emotion. In the future, this can be a point to ponder upon. Similarly, emojis can also denote emotion and as they are expressed in punctuation marks like :) for a happy face, and they are removed with the punctuation marks.

The GloVe model used for the vectorization is a pre trained vector model which used a collection of tweets for training. Reddit data can be different, and the vectorization model must be pre-trained on this data to get more accuracy.

Some of the papers that were published using this dataset have used transformer-based models to get better results, like BERT, and GPT. However, it would be interesting to see how they affect the data we have now, after the labels have merged.

5. References

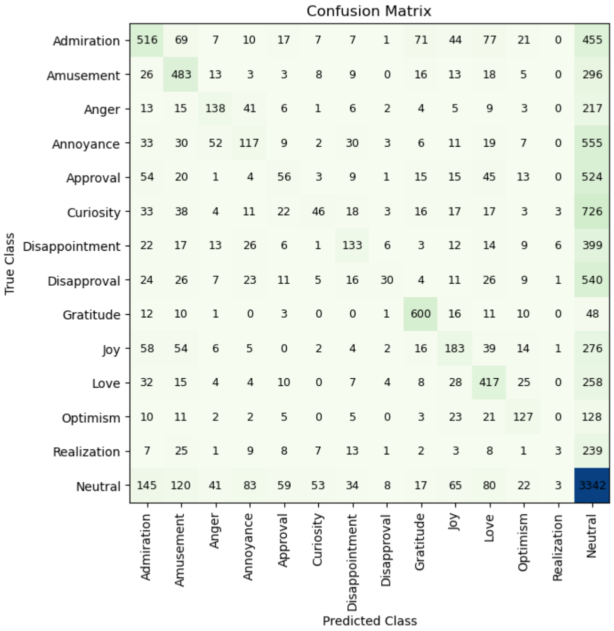
[1] Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G. and Ravi, S., 2020. GoEmotions: A dataset of fine-grained emotions. arXiv preprint arXiv:2005.00547.

[2] Olah, J., Baruah, S., Bose, D. and Narayanan, S., 2021. Cross domain emotion recognition using few shot knowledge transfer. arXiv preprint arXiv:2110.05021.

[3] Jeffrey, Pennington., Richard, Socher., Christopher, D., Manning. (2014). Glove: Global Vectors for Word Representation. 1532-1543. Available from: 10.3115/V1/D14-1162.

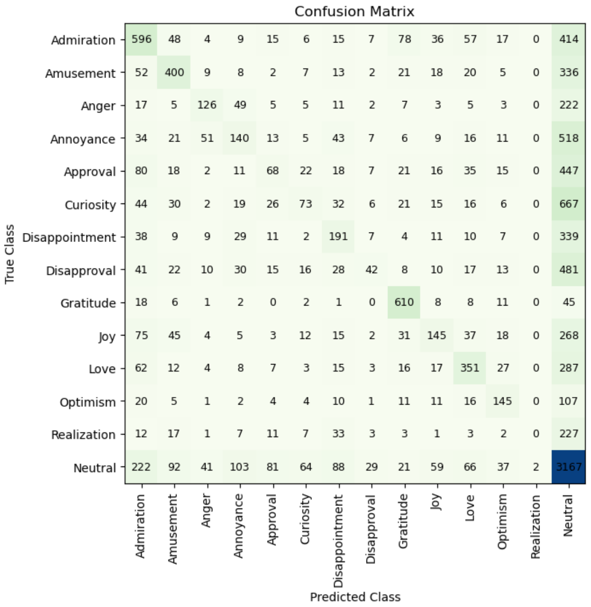
6. Appendix

6.1. Confusion matrix for Experiment 1



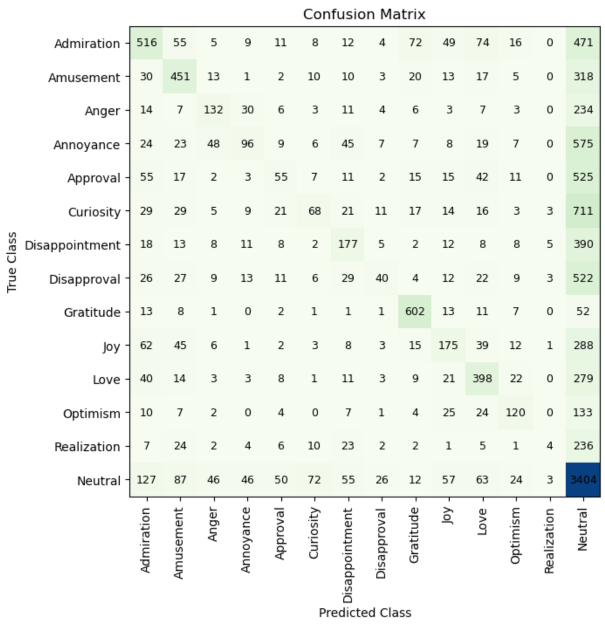
True Class \ Predicted Class	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	516	69	7	10	17	7	7	1	71	44	77	21	0	455
Amusement	26	483	13	3	3	8	9	0	16	13	18	5	0	296
Anger	13	15	138	41	6	1	6	2	4	5	9	3	0	217
Annoyance	33	30	52	117	9	2	30	3	6	11	19	7	0	555
Approval	54	20	1	4	56	3	9	1	15	15	45	13	0	524
Curiosity	33	38	4	11	22	46	18	3	16	17	17	3	3	726
Disappointment	22	17	13	26	6	1	133	6	3	12	14	9	6	399
Disapproval	24	26	7	23	11	5	16	30	4	11	26	9	1	540
Gratitude	12	10	1	0	3	0	0	1	600	16	11	10	0	48
Joy	58	54	6	5	0	2	4	2	16	183	39	14	1	276
Love	32	15	4	4	10	0	7	4	8	28	417	25	0	258
Optimism	10	11	2	2	5	0	5	0	3	23	21	127	0	128
Realization	7	25	1	9	8	7	13	1	2	3	8	1	3	239
Neutral	145	120	41	83	59	53	34	8	17	65	80	22	3	3342

Variation 2 Confusion matrix



True Class \ Predicted Class	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	596	48	4	9	15	6	15	7	78	36	57	17	0	414
Amusement	52	400	9	8	2	7	13	2	21	18	20	5	0	336
Anger	17	5	126	49	5	5	11	2	7	3	5	3	0	222
Annoyance	34	21	51	140	13	5	43	7	6	9	16	11	0	518
Approval	80	18	2	11	68	22	18	7	21	16	35	15	0	447
Curiosity	44	30	2	19	26	73	32	6	21	15	16	6	0	667
Disappointment	38	9	9	29	11	2	191	7	4	11	10	7	0	339
Disapproval	41	22	10	30	15	16	28	42	8	10	17	13	0	481
Gratitude	18	6	1	2	0	2	1	0	610	8	8	11	0	45
Joy	75	45	4	5	3	12	15	2	31	145	37	18	0	268
Love	62	12	4	8	7	3	15	3	16	17	351	27	0	287
Optimism	20	5	1	2	4	4	10	1	11	11	16	145	0	107
Realization	12	17	1	7	11	7	33	3	3	1	3	2	0	227
Neutral	222	92	41	103	81	64	88	29	21	59	66	37	2	3167

Variation 3 Confusion matrix



Confusion Matrix

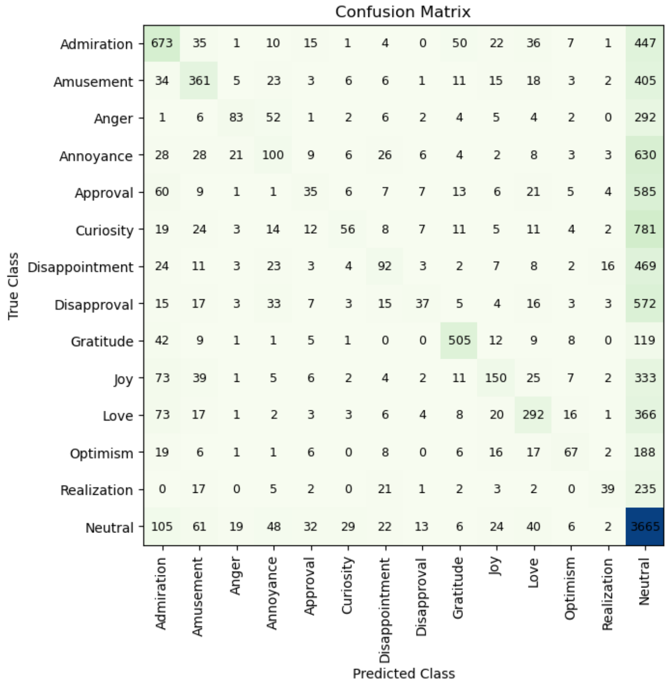
True Class

Predicted Class

	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	516	55	5	9	11	8	12	4	72	49	74	16	0	471
Amusement	30	451	13	1	2	10	10	3	20	13	17	5	0	318
Anger	14	7	132	30	6	3	11	4	6	3	7	3	0	234
Annoyance	24	23	48	96	9	6	45	7	7	8	19	7	0	575
Approval	55	17	2	3	55	7	11	2	15	15	42	11	0	525
Curiosity	29	29	5	9	21	68	21	11	17	14	16	3	3	711
Disappointment	18	13	8	11	8	2	177	5	2	12	8	8	5	390
Disapproval	26	27	9	13	11	6	29	40	4	12	22	9	3	522
Gratitude	13	8	1	0	2	1	1	1	602	13	11	7	0	52
Joy	62	45	6	1	2	3	8	3	15	175	39	12	1	288
Love	40	14	3	3	8	1	11	3	9	21	398	22	0	279
Optimism	10	7	2	0	4	0	7	1	4	25	24	120	0	133
Realization	7	24	2	4	6	10	23	2	2	1	5	1	4	236
Neutral	127	87	46	46	50	72	55	26	12	57	63	24	3	3404

Variation 4 Confusion Matrix

6.2. Experiment 2 Confusion matrix



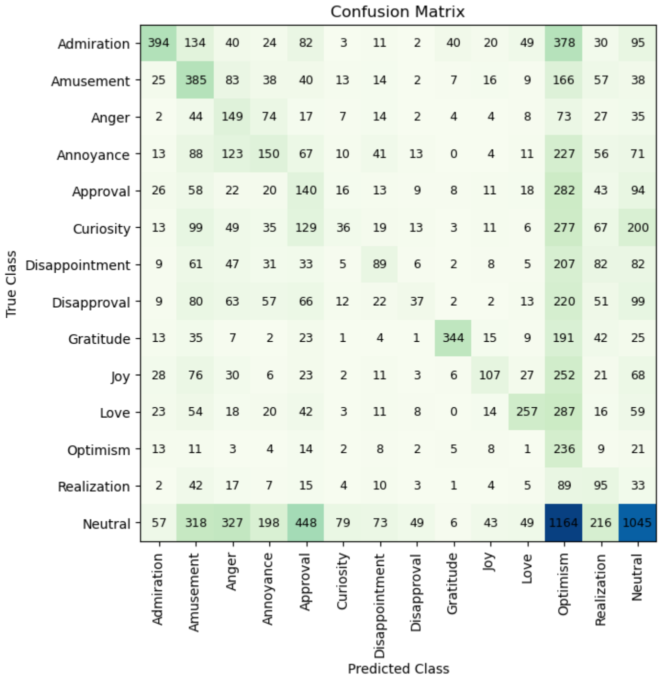
Confusion Matrix

True Class

Predicted Class

	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	673	35	1	10	15	1	4	0	50	22	36	7	1	447
Amusement	34	361	5	23	3	6	6	1	11	15	18	3	2	405
Anger	1	6	83	52	1	2	6	2	4	5	4	2	0	292
Annoyance	28	28	21	100	9	6	26	6	4	2	8	3	3	630
Approval	60	9	1	1	35	6	7	7	13	6	21	5	4	585
Curiosity	19	24	3	14	12	56	8	7	11	5	11	4	2	781
Disappointment	24	11	3	23	3	4	92	3	2	7	8	2	16	469
Disapproval	15	17	3	33	7	3	15	37	5	4	16	3	3	572
Gratitude	42	9	1	1	5	1	0	0	505	12	9	8	0	119
Joy	73	39	1	5	6	2	4	2	11	150	25	7	2	333
Love	73	17	1	2	3	3	6	4	8	20	292	16	1	366
Optimism	19	6	1	1	6	0	8	0	6	16	17	67	2	188
Realization	0	17	0	5	2	0	21	1	2	3	2	0	39	235
Neutral	105	61	19	48	32	29	22	13	6	24	40	6	2	3665

Variation 1 Confusion matrix



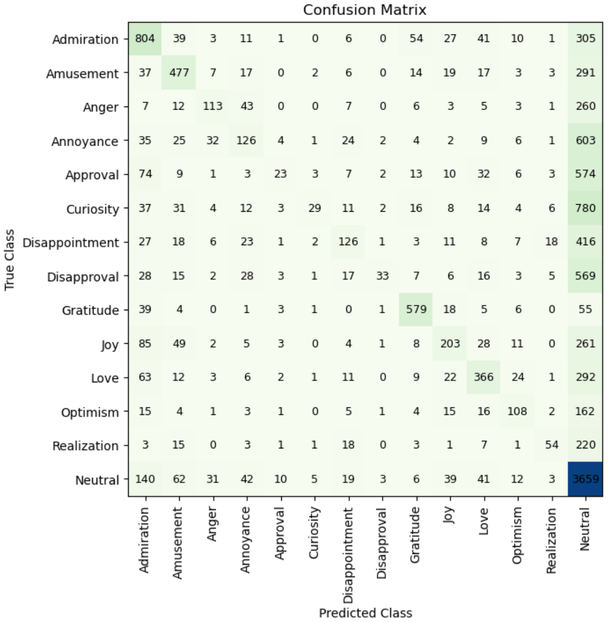
Confusion Matrix

True Class

Predicted Class

	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	394	134	40	24	82	3	11	2	40	20	49	378	30	95
Amusement	25	385	83	38	40	13	14	2	7	16	9	166	57	38
Anger	2	44	149	74	17	7	14	2	4	4	8	73	27	35
Annoyance	13	88	123	150	67	10	41	13	0	4	11	227	56	71
Approval	26	58	22	20	140	16	13	9	8	11	18	282	43	94
Curiosity	13	99	49	35	129	36	19	13	3	11	6	277	67	200
Disappointment	9	61	47	31	33	5	89	6	2	8	5	207	82	82
Disapproval	9	80	63	57	66	12	22	37	2	2	13	220	51	99
Gratitude	13	35	7	2	23	1	4	1	344	15	9	191	42	25
Joy	28	76	30	6	23	2	11	3	6	107	27	252	21	68
Love	23	54	18	20	42	3	11	8	0	14	257	287	16	59
Optimism	13	11	3	4	14	2	8	2	5	8	1	236	9	21
Realization	2	42	17	7	15	4	10	3	1	4	5	89	95	33
Neutral	57	318	327	198	448	79	73	49	6	43	49	1164	216	1045

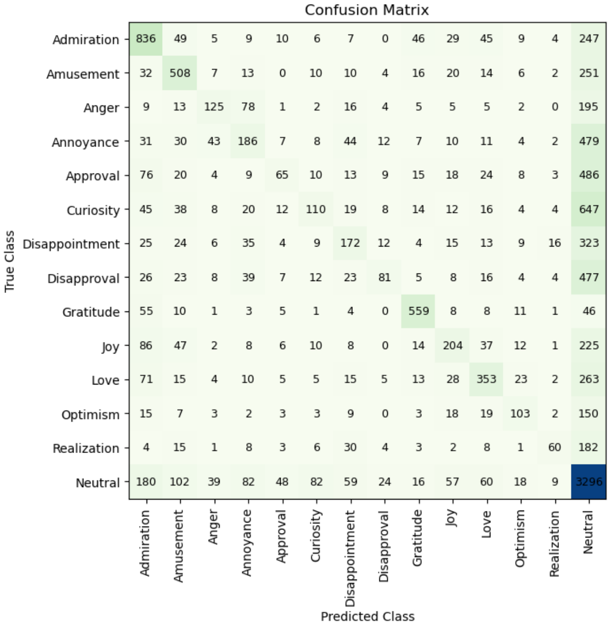
Variation 3 Confusion matrix



	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	804	39	3	11	1	0	6	0	54	27	41	10	1	305
Amusement	37	477	7	17	0	2	6	0	14	19	17	3	3	291
Anger	7	12	113	43	0	0	7	0	6	3	5	3	1	260
Annoyance	35	25	32	126	4	1	24	2	4	2	9	6	1	603
Approval	74	9	1	3	23	3	7	2	13	10	32	6	3	574
Curiosity	37	31	4	12	3	29	11	2	16	8	14	4	6	780
Disappointment	27	18	6	23	1	2	126	1	3	11	8	7	18	416
Disapproval	28	15	2	28	3	1	17	33	7	6	16	3	5	569
Gratitude	39	4	0	1	3	1	0	1	579	18	5	6	0	55
Joy	85	49	2	5	3	0	4	1	8	203	28	11	0	261
Love	63	12	3	6	2	1	11	0	9	22	366	24	1	292
Optimism	15	4	1	3	1	0	5	1	4	15	16	108	2	162
Realization	3	15	0	3	1	1	18	0	3	1	7	1	54	220
Neutral	140	62	31	42	10	5	19	3	6	39	41	12	3	3659

Variation 4 Confusion matrix

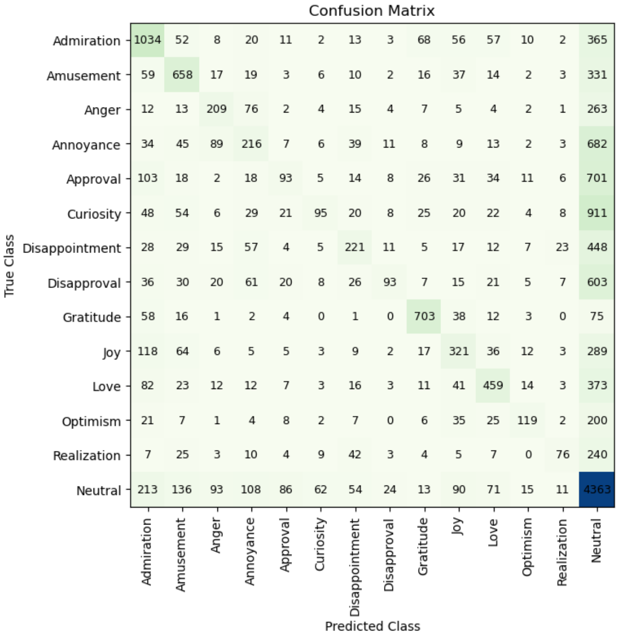
6.3. Experiment 3 – Confusion matrix



	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	836	49	5	9	10	6	7	0	46	29	45	9	4	247
Amusement	32	508	7	13	0	10	10	4	16	20	14	6	2	251
Anger	9	13	125	78	1	2	16	4	5	5	5	2	0	195
Annoyance	31	30	43	186	7	8	44	12	7	10	11	4	2	479
Approval	76	20	4	9	65	10	13	9	15	18	24	8	3	486
Curiosity	45	38	8	20	12	110	19	8	14	12	16	4	4	647
Disappointment	25	24	6	35	4	9	172	12	4	15	13	9	16	323
Disapproval	26	23	8	39	7	12	23	81	5	8	16	4	4	477
Gratitude	55	10	1	3	5	1	4	0	559	8	8	11	1	46
Joy	86	47	2	8	6	10	8	0	14	204	37	12	1	225
Love	71	15	4	10	5	5	15	5	13	28	353	23	2	263
Optimism	15	7	3	2	3	3	9	0	3	18	19	103	2	150
Realization	4	15	1	8	3	6	30	4	3	2	8	1	60	182
Neutral	180	102	39	82	48	82	59	24	16	57	60	18	9	3296

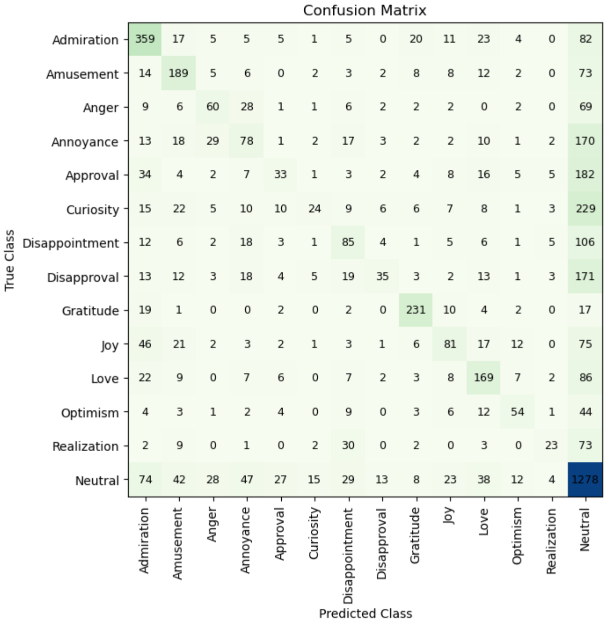
Variation 2 Confusion matrix

6.4. Experiment 4 – Confusion matrix



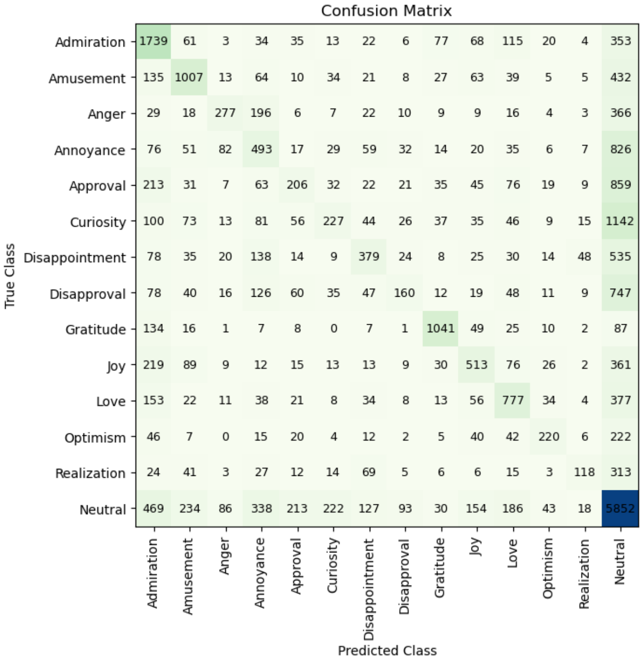
	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	1034	52	8	20	11	2	13	3	68	56	57	10	2	365
Amusement	59	658	17	19	3	6	10	2	16	37	14	2	3	331
Anger	12	13	209	76	2	4	15	4	7	5	4	2	1	263
Annoyance	34	45	89	216	7	6	39	11	8	9	13	2	3	682
Approval	103	18	2	18	93	5	14	8	26	31	34	11	6	701
Curiosity	48	54	6	29	21	95	20	8	25	20	22	4	8	911
Disappointment	28	29	15	57	4	5	221	11	5	17	12	7	23	448
Disapproval	36	30	20	61	20	8	26	93	7	15	21	5	7	603
Gratitude	58	16	1	2	4	0	1	0	703	38	12	3	0	75
Joy	118	64	6	5	5	3	9	2	17	321	36	12	3	289
Love	82	23	12	12	7	3	16	3	11	41	459	14	3	373
Optimism	21	7	1	4	8	2	7	0	6	35	25	119	2	200
Realization	7	25	3	10	4	9	42	3	4	5	7	0	76	240
Neutral	213	136	93	108	86	62	54	24	13	90	71	15	11	4363

Variation 2 Confusion matrix



	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	359	17	5	5	5	1	5	0	20	11	23	4	0	82
Amusement	14	189	5	6	0	2	3	2	8	8	12	2	0	73
Anger	9	6	60	28	1	1	6	2	2	2	0	2	0	69
Annoyance	13	18	29	78	1	2	17	3	2	2	10	1	2	170
Approval	34	4	2	7	33	1	3	2	4	8	16	5	5	182
Curiosity	15	22	5	10	10	24	9	6	6	7	8	1	3	229
Disappointment	12	6	2	18	3	1	85	4	1	5	6	1	5	106
Disapproval	13	12	3	18	4	5	19	35	3	2	13	1	3	171
Gratitude	19	1	0	0	2	0	2	0	231	10	4	2	0	17
Joy	46	21	2	3	2	1	3	1	6	81	17	12	0	75
Love	22	9	0	7	6	0	7	2	3	8	169	7	2	86
Optimism	4	3	1	2	4	0	9	0	3	6	12	54	1	44
Realization	2	9	0	1	0	2	30	0	2	0	3	0	23	73
Neutral	74	42	28	47	27	15	29	13	8	23	38	12	4	1278

Variation 3 Confusion matrix



	Admiration	Amusement	Anger	Annoyance	Approval	Curiosity	Disappointment	Disapproval	Gratitude	Joy	Love	Optimism	Realization	Neutral
Admiration	1739	61	3	34	35	13	22	6	77	68	115	20	4	353
Amusement	135	1007	13	64	10	34	21	8	27	63	39	5	5	432
Anger	29	18	277	196	6	7	22	10	9	9	16	4	3	366
Annoyance	76	51	82	493	17	29	59	32	14	20	35	6	7	826
Approval	213	31	7	63	206	32	22	21	35	45	76	19	9	859
Curiosity	100	73	13	81	56	227	44	26	37	35	46	9	15	1142
Disappointment	78	35	20	138	14	9	379	24	8	25	30	14	48	535
Disapproval	78	40	16	126	60	35	47	160	12	19	48	11	9	747
Gratitude	134	16	1	7	8	0	7	1	1041	49	25	10	2	87
Joy	219	89	9	12	15	13	13	9	30	513	76	26	2	361
Love	153	22	11	38	21	8	34	8	13	56	777	34	4	377
Optimism	46	7	0	15	20	4	12	2	5	40	42	220	6	222
Realization	24	41	3	27	12	14	69	5	6	6	15	3	118	313
Neutral	469	234	86	338	213	222	127	93	30	154	186	43	18	5852

Variation 4 confusion matrix