DEEP LEARNING FOR NATURAL LANGUAGE PROCESSING

Coding Assignment Report

Swetha Saseendran
CSE-C
185001183

Overview:

The aim of the project is to perform sentiment analysis on the given dataset of tweets and to study the existing sentiment analysis methods of Twitter data with different models and provide comparisons. The different techniques considered under the study include various ML and DL algorithms. (K-Means, Decision Tree, CNN, LSTM)

File descriptions:

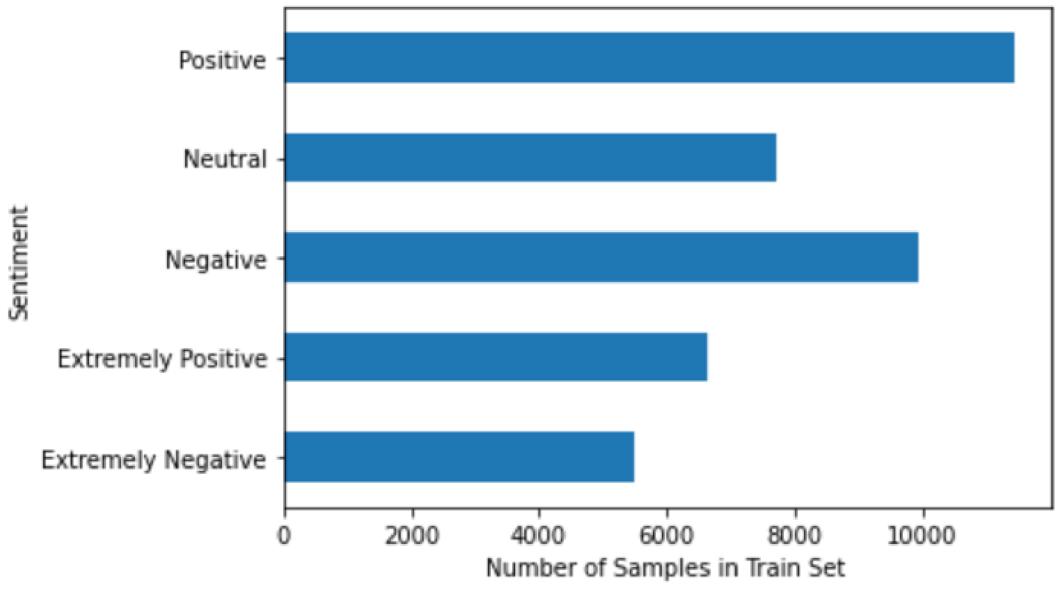
- **train.csv** - the training set
- **test.csv** - the test set

Data Fields:

- **UserName** - an anonymous id
- **ScreenName** - the id of screen
- **TweetAt** - the location of tweet
- **Tweet** - the tweet
- **Sentiment** - the sentiment

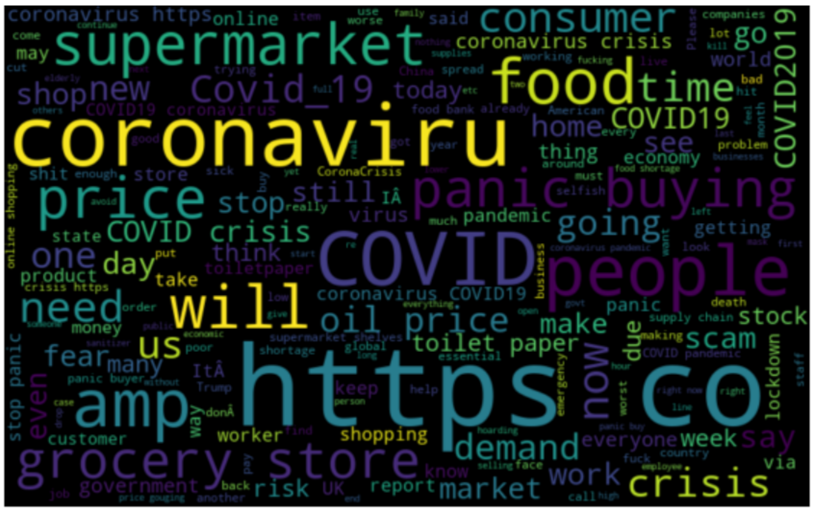
Exploratory Analysis:

Distribution of the Data:

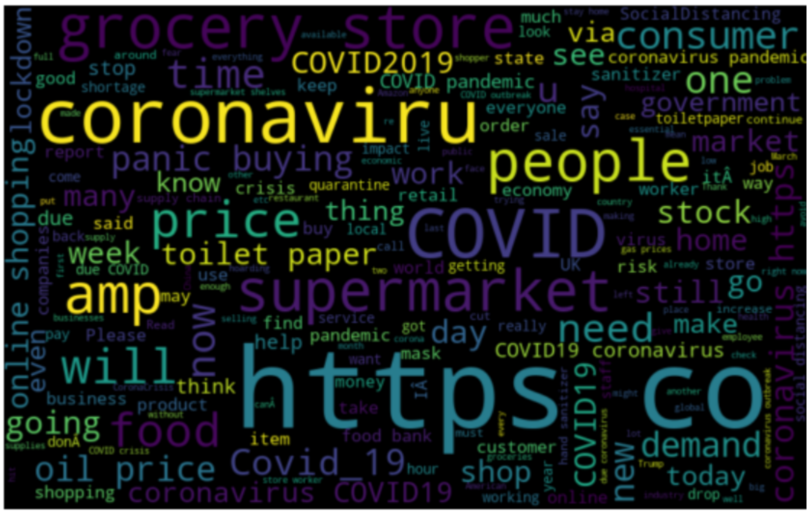


Word Map for various sentiments:

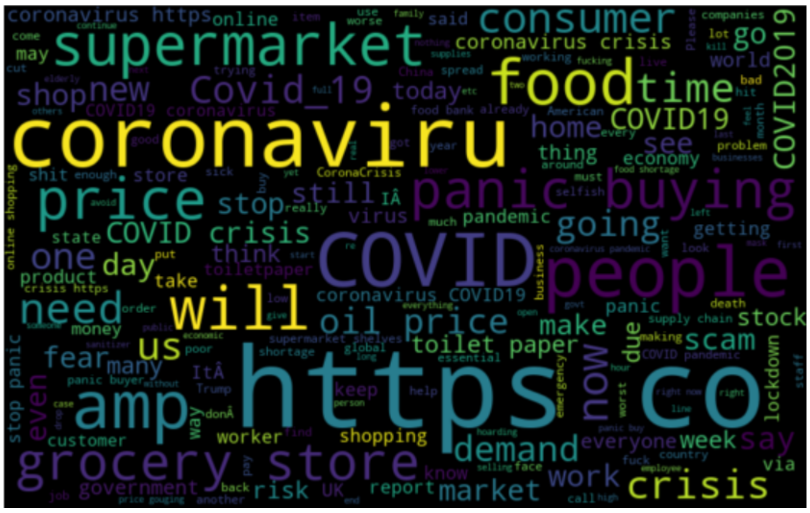
Extremely Negative:



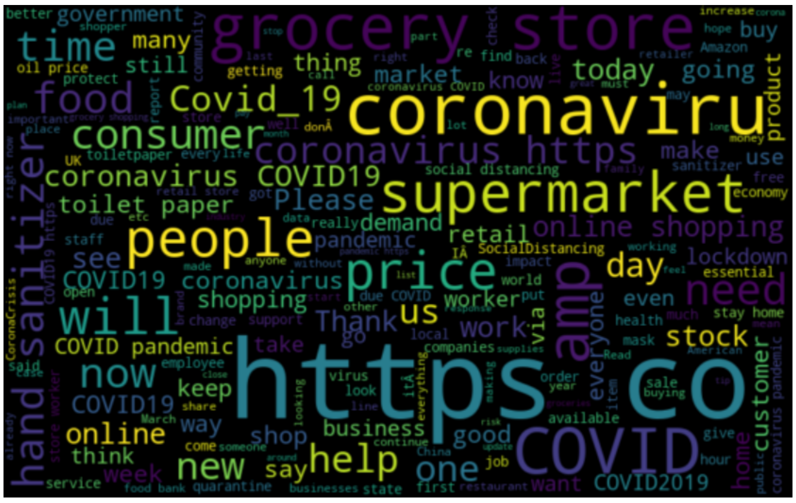
Negative:



Neutral:



Positive:



Extremely Positive:

Standard Steps Data Pre-Processing:

1. Remove unwanted characters including special characters, hashtags and http sites.
2. Tokenize the tweets.
3. Remove the stopwords.
4. Lemmatize the text.
5. Encoding the sentiments using One Hot Encoder.(Categorical Data)
    a. Extremely Negative (0)
    b. Extremely Positive (1)
    c. Negative (2)
    d. Neutral (3)
    e. Positive (4)
6. Word Embedding
    a. Word2Vec
    b. Text to sequence from Keras (for LSTM)
    c. One Hot Encoder (for Fine Tuned LSTM)

Methods Attempted:

★ **K-Means Clustering:**

K-Means Clustering Algorithm

- Number of clusters is taken to be 5 as there are 5 classes.
- Randomly assign each data point to a cluster
- Compute cluster centroids

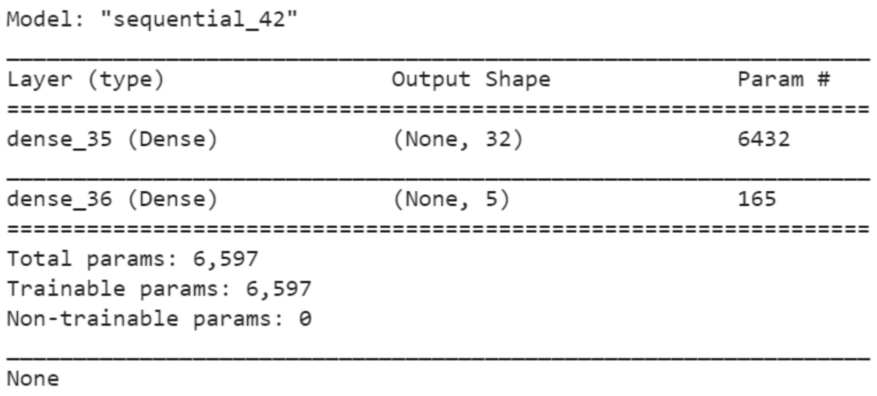
- Reassign each point to closest cluster centroid
- Re-compute cluster centroids
- Repeat 4 & 5 until no improvements are made

★ Decision Tree:

_____Decision trees are a classifier model in which each node of the tree represents a test on the attribute of the data set, and its children represent the outcomes. The leaf nodes represent the final classes of the data points. It is a supervised classifier model which uses data with known labels to form the decision tree and then the model is applied on the test data. For each node in the tree the best test condition or decision has P to be taken. We use the GINI factor to decide the best split. For a given node t, where p(j|t) is the relative frequency of class j at node t.

★ CNN:

CNN is used for Multi-Class classification and the model proposed is given below. Word2Vec is used to encode the tweets. Each tweet is encoded into a 200 length vector.



```
Model: "sequential_42"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_35 (Dense)             (None, 32)                6432
_________________________________________________________________
dense_36 (Dense)             (None, 5)                 165
=================================================================
Total params: 6,597
Trainable params: 6,597
Non-trainable params: 0
_________________________________________________________________
None
```

★ LSTM:

Aim to use the Long Short Term Memory (LSTM) architecture can be implemented using Keras for Multi-Class Text Classification with Scikit-Learn. The data is coded using One-Hot-Encoder. The model of the LSTM is given below.

```
Model: "sequential_7"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_6 (Embedding)      (None, 250, 100)          5000000
_________________________________________________________________
spatial_dropout1d_6 (Spatial (None, 250, 100)          0
_________________________________________________________________
lstm_6 (LSTM)                (None, 100)               80400
_________________________________________________________________
dense_6 (Dense)              (None, 5)                 505
=================================================================
Total params: 5,080,905
Trainable params: 5,080,905
Non-trainable params: 0
_________________________________________________________________
```

Quality Metrics and Comparison:

★ K-Means Clustering:

- Confusion Matrix

```
array([[ 774,   40,  139,   23,   80],
       [  72,  927,   77,   37,  217],
       [ 367,  185, 1133,   91,  230],
       [ 136,  121,  274,  814,  208],
       [ 188,  389,  364,  191, 1155]])
```

- Classification Report

	precision	recall	f1-score	support
0	0.50	0.73	0.60	1056
1	0.56	0.70	0.62	1330
2	0.57	0.56	0.57	2006
3	0.70	0.52	0.60	1553
4	0.61	0.51	0.55	2287
accuracy			0.58	8232
macro avg	0.59	0.60	0.59	8232
weighted avg	0.60	0.58	0.58	8232

★ Decision Tree:

- Classification Report

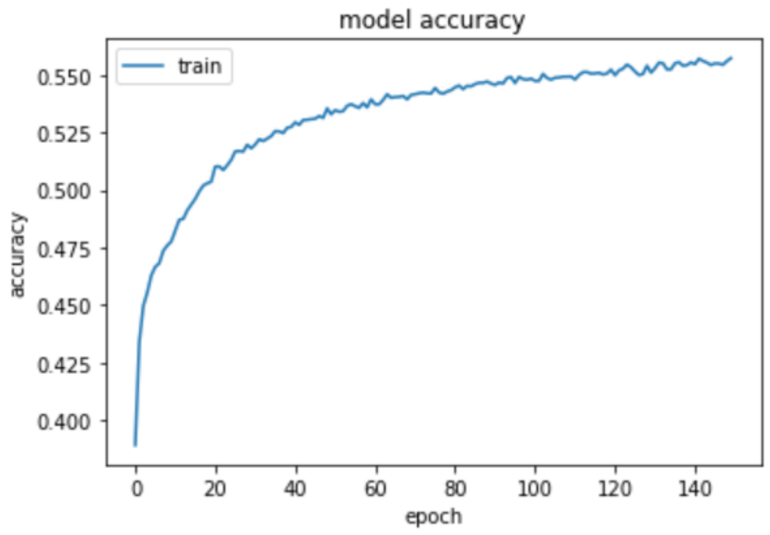
	precision	recall	f1-score	support
0	0.25	0.27	0.26	1056
1	0.28	0.28	0.28	1330
2	0.30	0.30	0.30	2006
3	0.32	0.31	0.31	1553
4	0.32	0.32	0.32	2287
accuracy			0.30	8232
macro avg	0.29	0.30	0.29	8232
weighted avg	0.30	0.30	0.30	8232

- Confusion Matrix

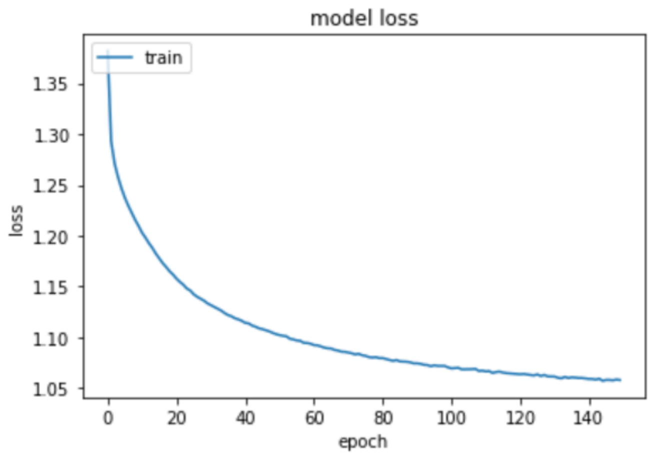
```
array([[288, 132, 288, 137, 211],
       [129, 375, 230, 181, 415],
       [307, 253, 593, 326, 527],
       [141, 178, 340, 479, 415],
       [275, 396, 499, 389, 728]])
```

★ <u>CNN :</u>

- Training Accuracy

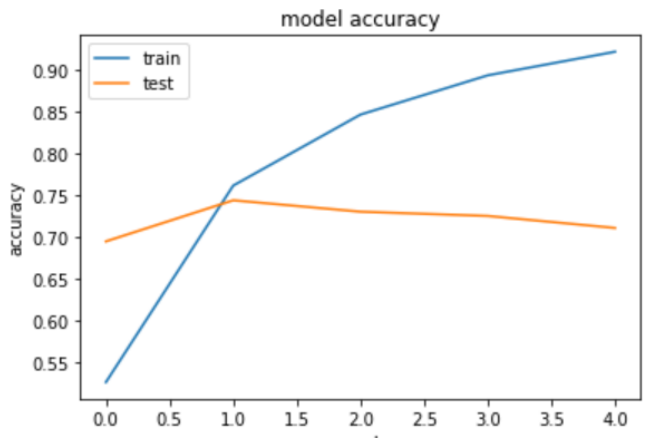


- Loss during training

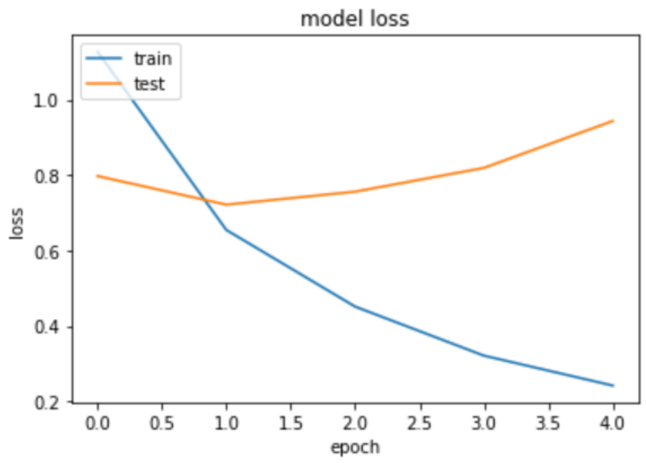


★ **LSTM:**

- **Training Accuracy**



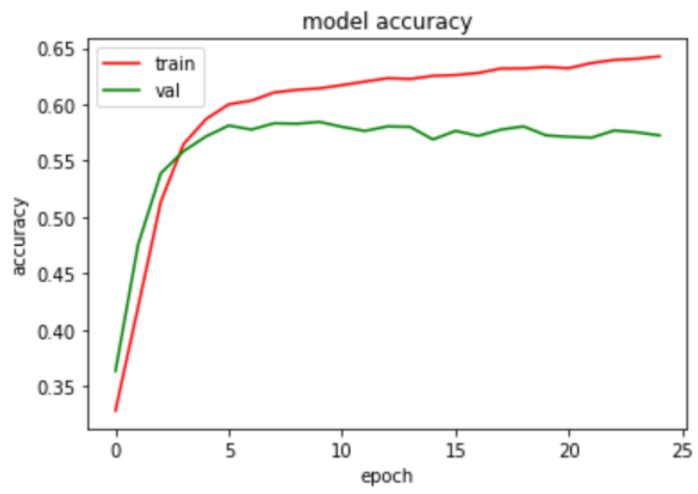
- **Loss during training**



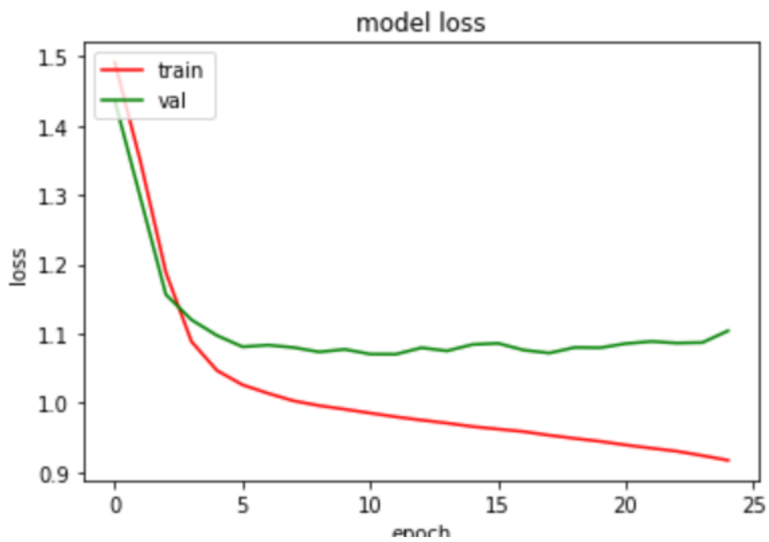
A model that is selected for its accuracy on the training dataset rather than its accuracy on an unseen test dataset is very likely to have lower accuracy on an unseen test dataset. Even though the model showed great accuracy for training data it seems to be over fitting. The loss increases on the validation set.
This can be prevented by early stopping or validation, which is done in the next model.

★ <u>**Fine Tuned LSTM:**</u>

- **Training Accuracy**



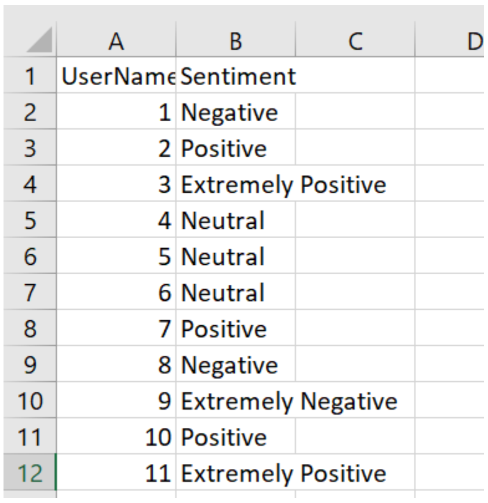
- **Loss during training**



Plot of learning curves shows a good fit if:

- The plot of training loss decreases to a point of stability.
- The plot of validation loss decreases to a point of stability and has a small gap with the training loss.

Output:



	A	B	C	D
1	UserName	Sentiment		
2	1	Negative		
3	2	Positive		
4	3	Extremely Positive		
5	4	Neutral		
6	5	Neutral		
7	6	Neutral		
8	7	Positive		
9	8	Negative		
10	9	Extremely Negative		
11	10	Positive		
12	11	Extremely Positive		

Conclusion:

- The dataset has been tested on various models and comparison has been drawn based on the quality metrics.
- From testing, it's evident that CNN and Fine Tuned LSTM produces better results with test data.