

SENTIMENT ANALYSIS OF NEWS HEADLINES USING SIMPLE TRANSFORMERS

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTER OF SCIENCE
IN
MATHEMATICS

Submitted by:

Anurag Singh
(2K19/MSCMAT/02)

Under the supervision of
Dr. Goonjan Jain



DEPARTMENT OF APPLIED MATHEMATICS
DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi – 110042

MAY, 2021

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi - 110042

CANDIDATE'S DECLARATION

I, (Anurag Singh), 2K19/MSCMAT/02, student of M.Sc. Mathematics, hereby declare that the project Dissertation titled “**Sentiment Analysis of News Headlines Using Simple Transformers**” which is submitted by me to the Department of Applied Mathematics, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Science, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship, or other similar title or recognition.

Place: Delhi

Date: 25 May 2021

Anurag Singh
Anurag Singh

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi – 110042

CERTIFICATE

I hereby certify that the Project Dissertation titled “**Sentiment Analysis of News Headlines Using Simple Transformers**” which is submitted by Anurag Singh, 2K19/MSCMAT/02 of Department of Applied Mathematics, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Science, is a record of the project work carried out by the students under my supervision. To the best of my knowledge, this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date: 25 May 2021



Dr. Goonjan Jain

SUPERVISOR

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi – 110042

ABSTRACT

With the rate at which the data is being generated, it is vital to use it and get some insights from it. When reporting on events, news expresses the opinions of news entities like people, locations, and things. In this paper, we obtain the sentiment of the news headlines using a new technique called transformers, in particular simple transformers that have been a significant advancement in the natural language processing field. Sentiment Analysis has been performed using the four transformers model. These models are pre-trained on extensive data, and we have fine-tuned them by training them on our own news headlines dataset.

For our sentiment analysis task, classification models (specific simple transformer model) are used to classify news headlines as negative, neutral, positive. The idea behind taking four different models that are Bidirectional Encoder Representations from Transformers (BERT) base-cased [1], Robustly optimized BERT approach (RoBERTa) base [1], Distilled BERT (DistilBERT) base-cased [1], XLNet base-cased [1], is the different dataset on which they are pre-trained, parameters used by them, and different method used by them, which improve their performance significantly in comparison to different machine learning classifiers and prior deep learning models. The model that performed the best is bert-base-cased with Matthews Correlation Coefficient (MCC) score of 90.1%, an F1 score of 93.6%, and an Accuracy score of 93.6%.

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi – 110042

ACKNOWLEDGEMENT

I want to express my sincere gratitude to my supervisor, Dr Goonjan Jain, for her guidance and assistance in completing this project Dissertation, which allowed me to conduct extensive research and learn about many new things. Second, I want to thank all the researchers whose research papers are mentioned in the reference section, which aided in completing this Dissertation.

CONTENTS

CANDIDATE’S DECLARATION	ii
CERTIFICATE	iii
ABSTRACT	iv
ACKNOWLEDGEMENT	v
CONTENTS	vi
LIST OF FIGURES	vii
LIST OF TABLES	viii
CHAPTER 1 INTRODUCTION	1
CHAPTER 2 RELATED WORK	2
CHAPTER 3 PROPOSED WORK	4
3.1 DATA COLLECTION AND PRE-PROCESSING	5
3.2 LABELLING AND SPLITTING DATASET	6
3.3 MODEL TRAINING AND EVALUATION	6
3.4 APPLICATION OF SOFTMAX FUNCTION	6
3.5 PREDICTION OF UNLABELLED DATA	8
CHAPTER 4 RESULTS AND DISCUSSION	8
4.1 MEASUREMENTS	10
4.2 OUTPUT AND VISUALIZATIONS	12
4.3 CODE AND DATASET SNIPPETS	16
CONCLUSION	18
REFERENCES	22

LIST OF FIGURES

Fig. 3.1 Workflow for the news headlines dataset

Fig. 3.4.1 Softmax function

Fig. 3.4.2 Model output matrix example

Fig. 4.1 Basic RNN Structure

Fig. 4.2 Transformer Architecture

Fig. 4.2.1 mcc score vs global steps

Fig. 4.2.2 eval loss vs global steps

Fig. 4.2.3 train loss vs global steps

Fig. 4.2.4 ROC curve

Fig. 4.2.5 Confusion matrix for BERT model

Fig. 4.2.6 Confusion matrix for ROBERTa model

Fig. 4.2.7 Confusion matrix for DistilBERT model

Fig. 4.2.8 Confusion matrix for XLNet model

LIST OF TABLES

Table 4.2.1 Output table for model metrics

CHAPTER 1 INTRODUCTION

Opinions or Sentiments are primary to every human exercise and play a key role in influencing our behavior. Sentiment analysis is a method to extricate sentiments or opinions from the text data regarding positive, negative, or neutral opinions expressed in natural language. Sentiment analysis is applied in various fields like businesses, organizations, politics, sports, etc., and with the increasing amount of data, it has become a necessity nowadays. The sentiment analysis is conducted at four levels: word level, sentence level, document level, and aspect level [2]. According to [3], the Obama administration used sentiment analysis to measure public opinion before a presidential election. With the increase in news sources and access to the internet, the volume of news is increasing. As per [4], most people evaluate news reports by scanning the headlines rather than reading the whole news. Readers are more likely to use online news outlets due to two main factors: interactivity and immediacy [5].

Some of the approaches of extracting sentiment or opinion from the text are unsupervised learning [6], supervised learning [6], and semi-supervised learning [6]. When we train a machine using the data neither labeled nor classified, the first approach allows the algorithm to learn from the data without supervision. In the second approach, we train a machine using labelled data, and after that, the machine is given a new set of unknown data. The third approach is a combination of the first two approaches in which a machine is trained upon a combination of labelled and unlabeled data, therefore cutting down the labelled data expense.

In this paper, Sentiment analysis is done based on the news headlines, instead using the entire news article. The method proposed here starts with the data collection and pre-processing. Labels for the news headlines were assigned as 0(Negative), 1(Neutral), 2(Positive) using the natural language toolkit's built in Vader sentiment analyzer that ranks a text as negative, neutral, positive based on the sentiments. We used Transformer models, specifically simple transformers, which are exceptional in advancing Natural Language Processing.

Advances in both model design and model pre-training have fueled recent advances in natural language processing. Transformer architectures have made it easier to create higher-capacity models, and pretraining has enabled this capacity to be used for a broad range of tasks [7]. Simple transformer is a library built on Hugging Face's Transformers library [8]. Every model is packed with functions and features that are most appropriate for the job at hand. It lets you train, test, and predict in a short span of time. They are designed with a specific natural language task in mind [8]. Simple transformers models can be used in the following steps: (1) initializing a task-specific model [8], (2) training the model using the training data [8], (3) evaluate the model using the testing data [8], (4) making predictions on unlabelled data [8]. The main differences between models will probably be differences in the format of input/output data as well as any task-specific features/configuration choices [8].

CHAPTER 2 RELATED WORK

Various researches have contributed to news sentiment analysis using various approaches. A brief review of work done formally on sentiment analysis is given in this section.

According to [9] proposed a research and compared different classification and text mining algorithms for analysing sentiment in newspapers. Maximum Entropy, Naive Bayes, Decision Tree, winnow, and C 4.5, were the classifiers included. Instead of editorials, reviews, or blogs news articles were chosen as they contain less emotionally loaded texts than the former. Their results reveal that the Naive Bayes classifier outperforms all others by a wide margin.

As per [10] investigated possible strategies used by web news corporations to design their news headlines. They used 69,907 news headlines from four major news organizations. They designed the methodology to use headlines as a key artifact because they are critical in gaining interest in online news. Their methodology employed sentiment analysis to understand inherent aspects of headlines in order to predict news popularity dependent on URL shortening service. They discovered that the headline's sentiment is closely linked to the news's popularity.

According to [11] presented an approach for awarding scores to every different entity in the text corpus that indicates positive or negative opinion. The approach included a component of sentiment identification, that associated expressed sentiments to each relevant entity, as well as a sentiment aggregation and scoring component that scored each and every entity relative to others of the same class [11]. For sentiment analysis from news and blogs, the news dimensions considered were crime, general, health, business, media, sports, and politics [11]. The following were two patterns that were tracked over time using sentiment score: (1) Polarity: positive or negative sentiment linked with the entity [11], and (2) the amount of sentiment an entity obtained [11].

According to [7] transformer architecture is especially well suited to pretraining on massive text corpora, resulting in significant improvements in accuracy on downstream tasks

such as language understanding [12], coreference resolution [13], text classification [14], common-sense inference [15], summarization [16], and machine translation [17] among others.

The effectiveness of transformers for varied NLP tasks motivated us to implement transformers-based sentiment analysis.

CHAPTER 3 PROPOSED WORK

This paper will analyze and introduce transformers using a simple transformers library in the easiest way possible. We will use the classification model from the simple transformers library to conduct our sentiment analysis of news headlines. The simple transformer models that are used here are Bidirectional Encoder Representations from Transformers (BERT) base-cased [1], Robustly optimized BERT approach (RoBERTa) base [1], Distilled BERT (DistilBERT) base-cased [1], XLNet base-cased [1]. Each of these models is pre-trained on large text data. The techniques and the volume of data used by these models are what sets them apart. The techniques used by different models are: (1) BERT uses masked language model, and next sentence prediction [1], (2) RoBERTa replaces the next sentence prediction task in bert with dynamic masking [1], (3) DistilBERT is a distilled version of bert using half of the number of parameters of the bert [1], (4) XLNet uses permutation language modelling [1].

The methodology for this experiment has been presented in Fig. 3.1 it consists of 6 steps, starting with data collection and pre-processing. The News headlines dataset has been used here for the experiment. The steps have been explained below in detail.

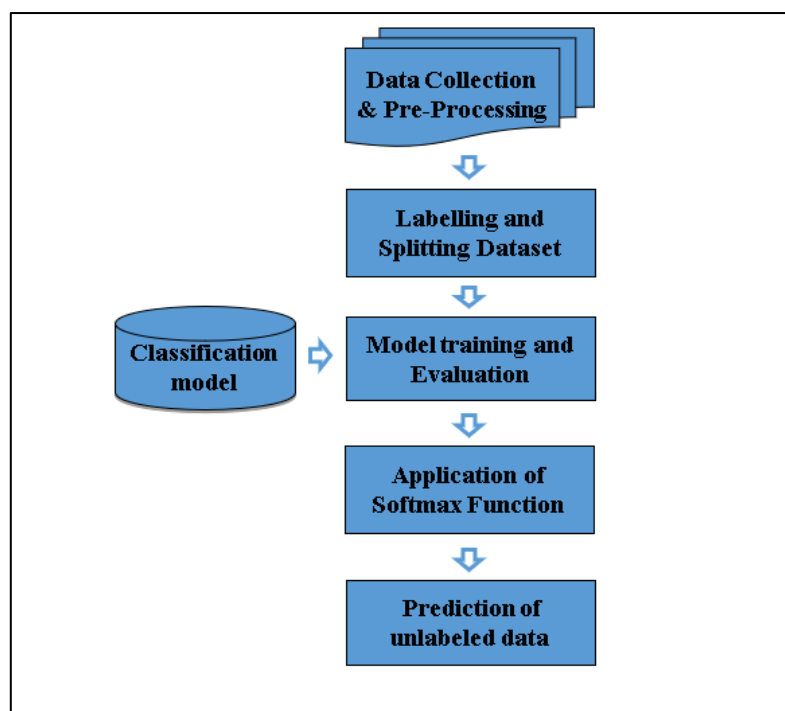


Figure 3.1 Workflow for the news headlines dataset

3.1 DATA COLLECTION AND PRE-PROCESSING

The news headlines dataset used here were collected from different news websites worldwide by web-scraping through a Python script and Python's newspaper package. News headlines dataset and code is uploaded at github. This dataset comprises a total of 33,801 news headlines. The news headlines belong to the business, sports, health, politics, tech, entertainment, education, science, lifestyle, environment, economy, crime, blogs, automobiles, travel, economics.

News headlines in the dataset do not require much pre-processing as most of the work is handled by classification models of simple transformer. Other than this, some of the pre-processing tasks done here are removing punctuation marks, numbers, special symbols, NA-values.

3.2 LABELLING AND SPLITTING DATASET

The process of supervised sentiment analysis requires data to be labelled. The natural language toolkits built-in Vader sentiment analyzer used here provides a compound value that ranges from -1 (Extremely Negative) to 1 (Extremely positive); with some experimentation and testing, we consider texts with a compound value less than 0 as negative, greater than 0 as positive, and equal to 0 as neutral. After labelling, data was split in a ratio of 80% (training data) and 20% (testing data) using the python library sklearn model selection train test split with the shuffle parameter as true and random state parameter 10.

3.3 MODEL TRAINING AND EVALUATION

Model training with a good range of hyperparameter values are critical to achieving a good outcome. Machine learning models can be highly sensitive to the hyperparameters used to train them [8]. Although large models, such as transformers, will work well over a broader range of hyperparameters. All the four classification models discussed above are initialized, and hyperparameters values are given to them after this training data is used for training each model with early stopping set to true used as one of the hyperparameters to prevent model overfitting [8]. Once the training part is over, the model evaluation performed on the test data. In the results and discussion section, the evaluation results for each model are listed.

3.4 APPLICATION OF SOFTMAX FUNCTION

After getting results on evaluation, the softmax function used to get the probabilities of each class of the news. The softmax function is another kind of activation function used in neural computation [18]. It converts a list of N real values to a list of N real values that add up to 1

[19]. The softmax converts input values that are zero, positive, negative, or more than one into values between 0 and 1, allowing them to be represented as probabilities [19]. Since several multi-layer neural networks generate real-valued scores that are difficult to scale and interact with, the softmax layer can be used in this situation because it converts the scores to a normalised probability distribution that can be presented to the user or used as input to other systems [19]. Softmax function working [[20], Fig. 3.4.1].

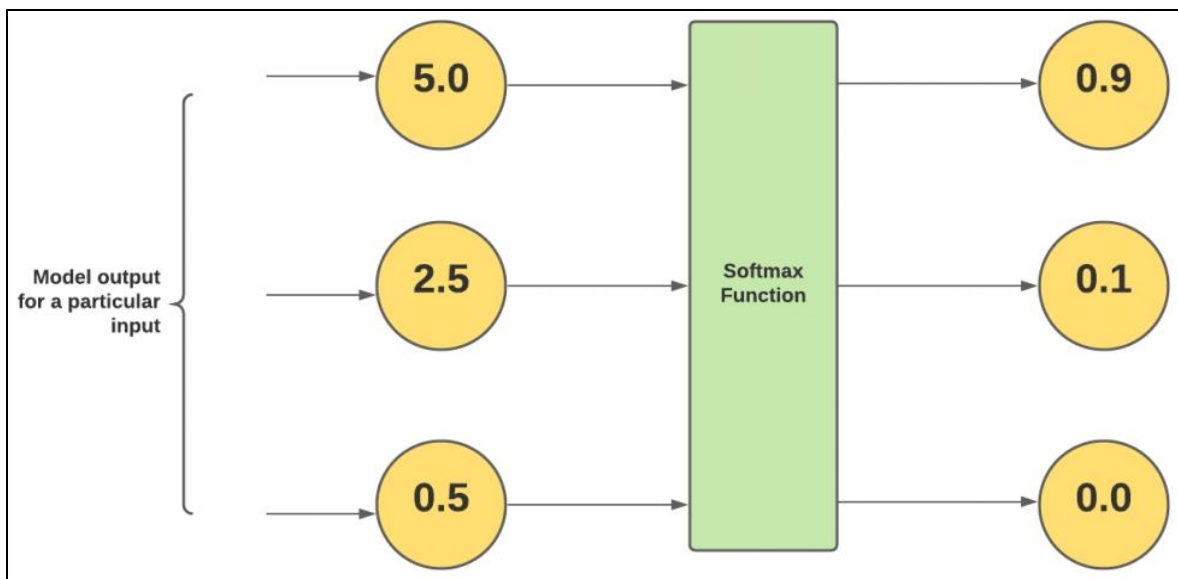


Fig. 3.4.1 Softmax function

The formula used to compute softmax function is:

$$f(x_i) = \frac{e^{x_i}}{\sum_{j=1}^N e^{x_j}} \quad (3.4.1)$$

The mathematical definition of the terms and symbols in [[18], eq. 3.4.1]: (1) x_i 's are components of softmax function's input list/vector, and they can take positive, negative, zero, real, positive value. (2) $\exp(x_i)$ is an exponential function that is applied to each component

of the input vector. It yields a positive value greater than zero, which is very low when the input is less than zero and very high when the input is large [19]. (3) $\sum_{j=1}^N \exp(x_j)$ this term ensures all the function's output values add up to 1 by normalization; [19]. (4) N is the number of classes like in this paper there are 3 classes negative, neutral, positive.

The model output matrix obtained in the evaluation step fed to the softmax function, where the function returns probability values for each class, with the main target class having the highest probability. A sample model output matrix in Fig. 3.4.2:

```
[ [-3.12695312  3.45898438 -0.77490234]
 [ 3.50585938 -1.58789062 -1.60058594]
 [ 6.69921875 -2.97460938 -2.28515625]
 ...
 [-2.77734375 -3.35546875  5.37109375]
 [-2.92382812  5.4609375  -2.64257812]
 [-3.09960938  5.5234375  -2.52929688]]
```

Fig. 3.4.2 Model output matrix example

3.5 PREDICTION OF UNLABELLED DATA

In the real world, we do not know what an accurate label is, so we use the predict function, which is nearly the same as the evaluate function, to make predictions on unlabelled data. It accepts a text list and returns a list of predictions, and a model output matrix.

CHAPTER 4 RESULTS AND DISCUSSION

The comparative analysis generated using the developed news headlines dataset and the

implementation of four different classification models were plotted using a line graph, a ROC curve, a confusion matrix, and the analysis is done.

A python library wandb used to visualise each classification model at different stages wherever required. As per [21], “Wandb, a framework-agnostic experiment tracking tool for machine learning”. It was created to make it easier for people to build machine learning models and save them.

Different machine learning classifiers can be used to perform sentiment analysis an application of text classification. With the amount of data generated, we are bound to see developments, which leads to the machine learning subfield of deep learning. Developments in deep learning resulted in neural network architectures such as convolutional neural networks (CNN) [22], and recurrent neural networks (RNN and LSTM) [22], showing a significant increase in efficiency while tackling natural language tasks. Transfer learning is defined as a methodology in which a deep learning model is trained on a massive dataset to handle related tasks on another dataset [22]. These deep learning models are referred to as pre-trained models. Since sequence modelling tasks like text classification, machine translation, etc., cannot be handled by traditional machine learning models and neural networks, this led to RNNs and CNNs. RNN structure [[22], Fig. 4.1].

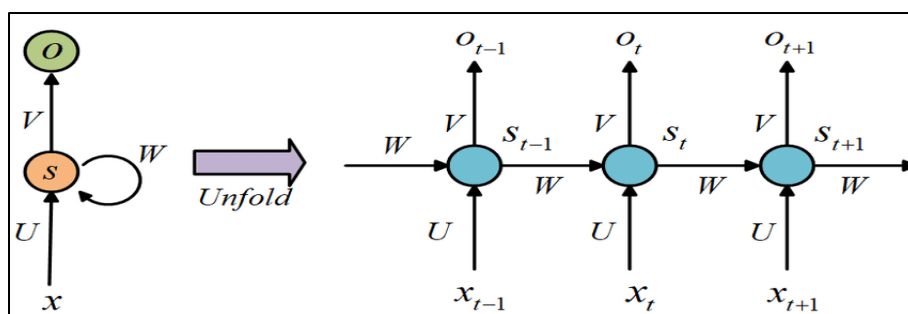


Fig. 4.1 Basic RNN Structure

RNNs can take one token at a time as input since they cannot be parallelized, thus consuming a significant amount of time while training on massive datasets [22]. This issue prompted the introduction of transfer learning in natural language processing, which resulted in a wide variety of task-specific transformer models. One of the benefit of the transformers is that they accept a complete sequence as input rather than processing token by token [22], and therefore the training can be accelerated by using GPU's. Transformers [[22], Fig. 4.2].

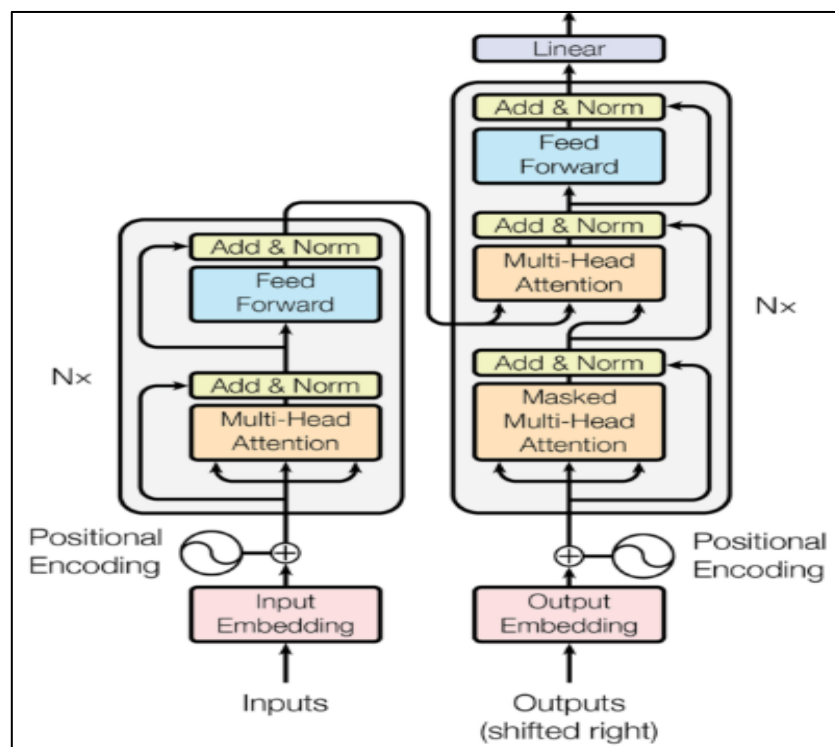


Fig. 4.2 Transformer Architecture

4.1 MEASUREMENTS

- **Matthews Correlation Coefficient (MCC):** It is a measure of quality of multiclass classification. It takes into consideration all the blocks of the Confusion Matrix in its formula as seen in [[23], eq. 4.1.1]. Its value lies between -1 to +1 with former indicating poor model and later indicating perfect model. This is shown in Fig. 4.2.1.

$$MCC = \frac{t_p * t_n - f_p * f_n}{\sqrt{(t_p + f_p)(t_p + f_n)(t_n + f_p)(t_n + f_n)}} \quad (4.1.1)$$

t_p , t_n , f_p , and f_n are true positives, true negative, false positive, false negative.

- **F1-Score:** It is defined as the weighted average of the precision and recall with best value as 1 and worst value as 0. Precision and recall have equal contribution to the F1-score as seen in [[23], eq. 4.1.2].

$$F1 - Score = \frac{2 * (Precision * Recall)}{(Precision + Recall)} \quad (4.1.2)$$

- **Accuracy:** It is calculated by dividing the total number of correct predictions by the total number of predictions.
- **Eval Loss:** It is the evaluation loss occurred during evaluation of the model on the dataset. This is shown in Fig. 4.2.2.
- **Train loss:** It is the training loss occurred during training of the model on the dataset. This is shown in Fig. 4.2.3.
- **Global steps:** It is a global step in processing that gets incremented everytime Wandb.log is called. Wandb.log is used to log data from runs [21].
- **Confusion matrix:** It is a $k * k$ matrix used for classification model performance analysis; it compares actual values to values predicted by the model. Here k represents the number of target classes. This is shown in Figures 4.2.5, 4.2.6, 4.2.7, and 4.2.8.

4.2 OUTPUT AND VISUALIZATIONS

Table 4.2.1 Output table for model metrics

Classification Models	Model Metrics			
	MCC Score	F1- Score	Accuracy	Eval Loss
BERT base cased	90.1%	93.6%	93.6%	38.8%
ROBERTa base	89.4%	93.2%	93.2%	40.0%
DistilBERT base cased	89.0%	92.9%	92.9%	39.5%
XLNet base cased	87.5%	91.9%	91.9%	43.9%

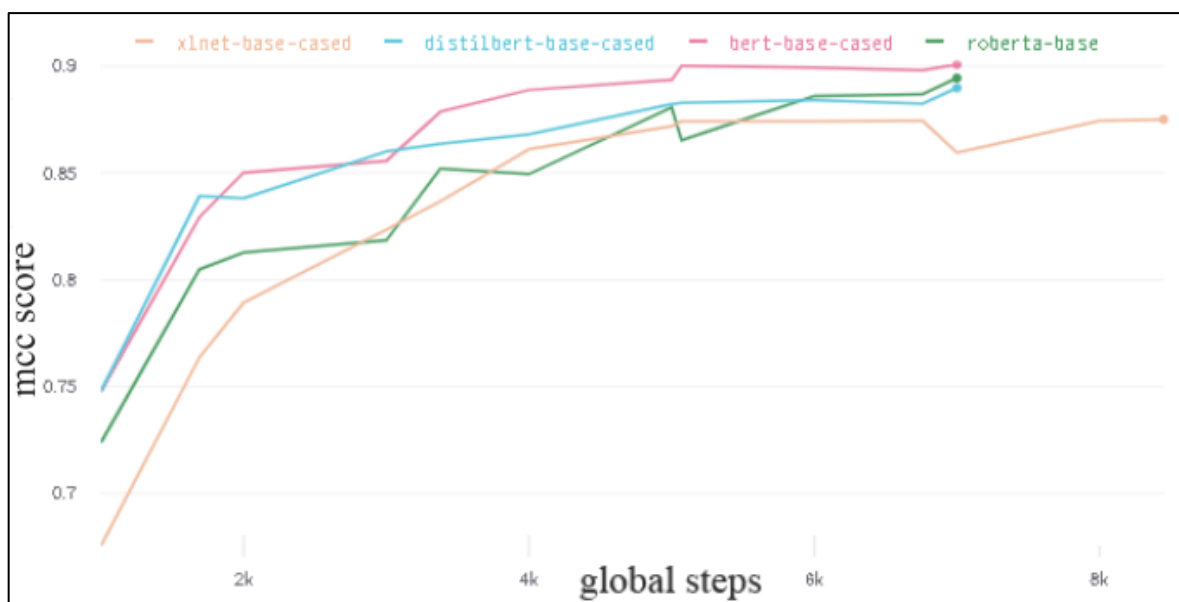


Fig. 4.2.1 mcc score vs global steps

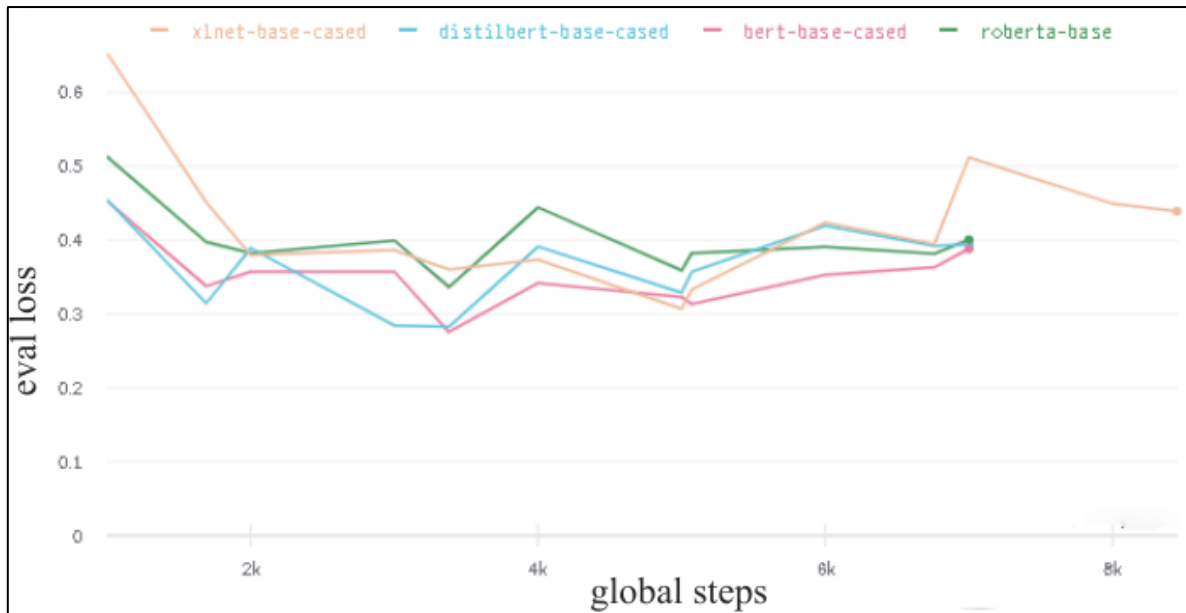


Fig. 4.2.2 eval loss vs global steps

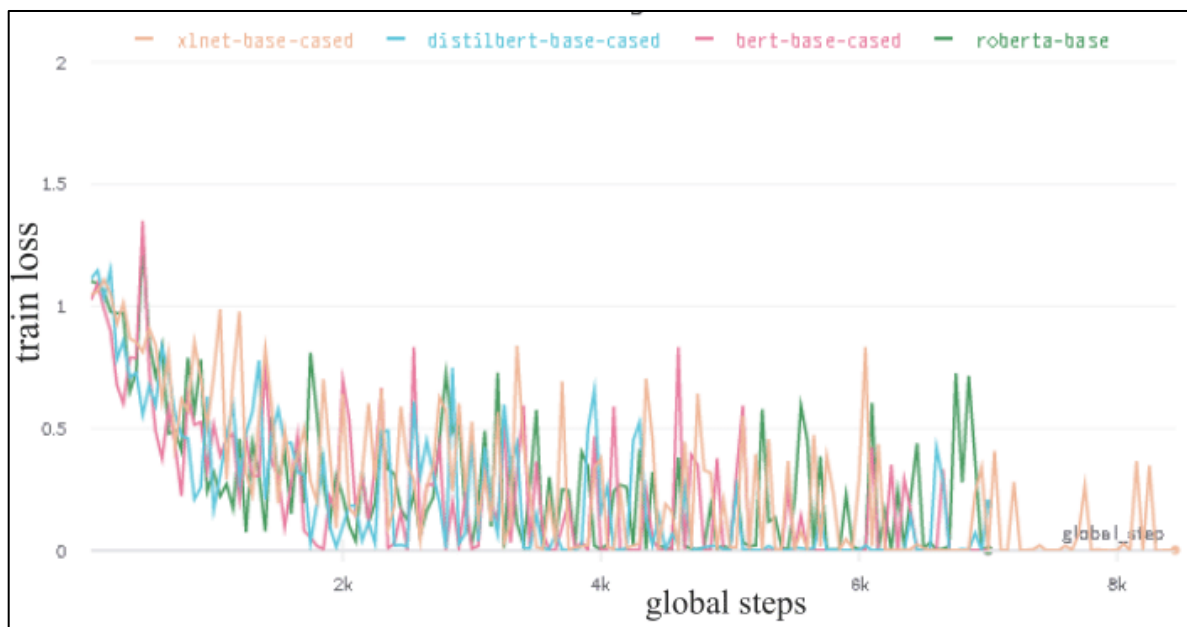


Fig. 4.2.3 train loss vs global steps

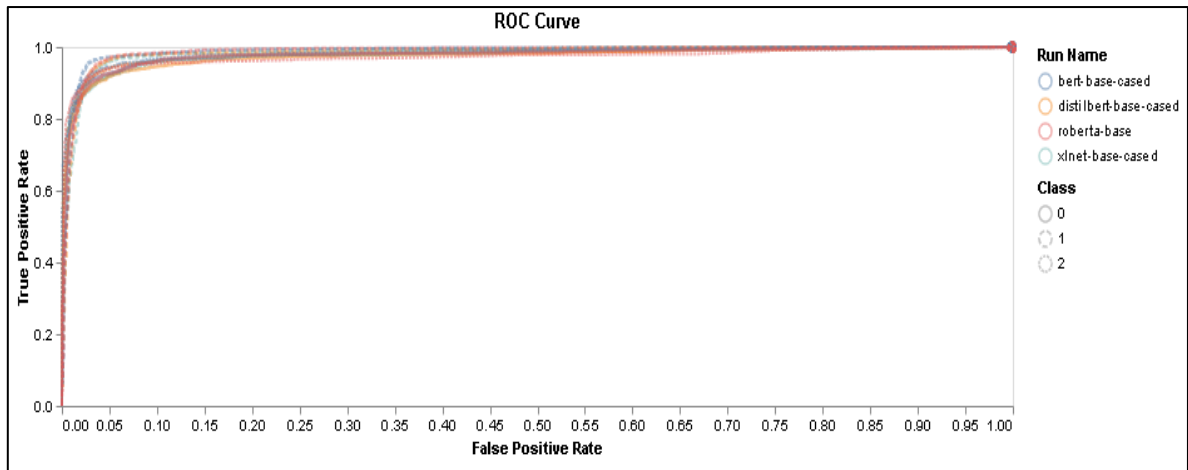


Fig. 4.2.4 ROC curve

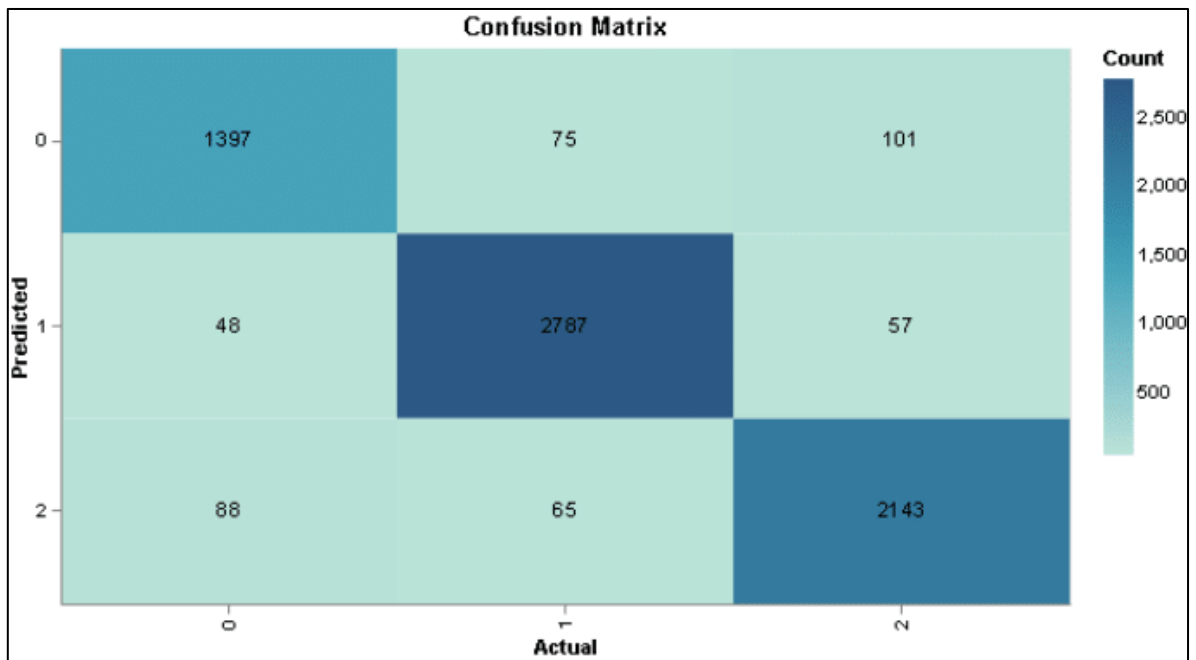


Fig. 4.2.5 Confusion matrix for BERT model

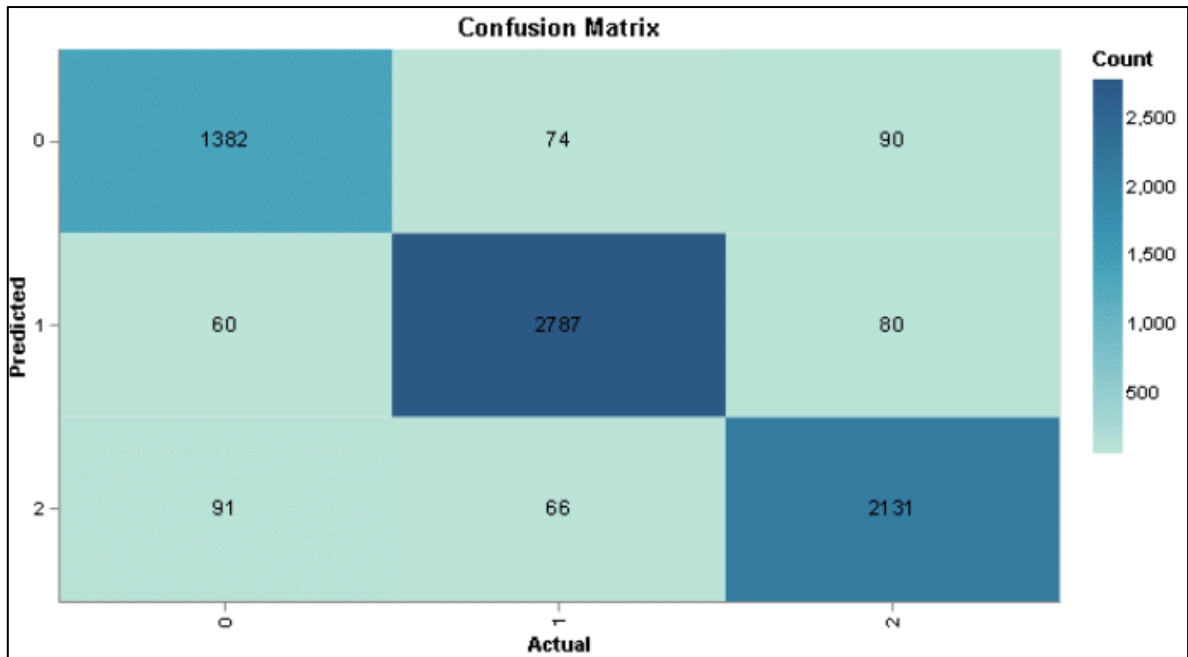


Fig. 4.2.6 Confusion matrix for ROBERTa model

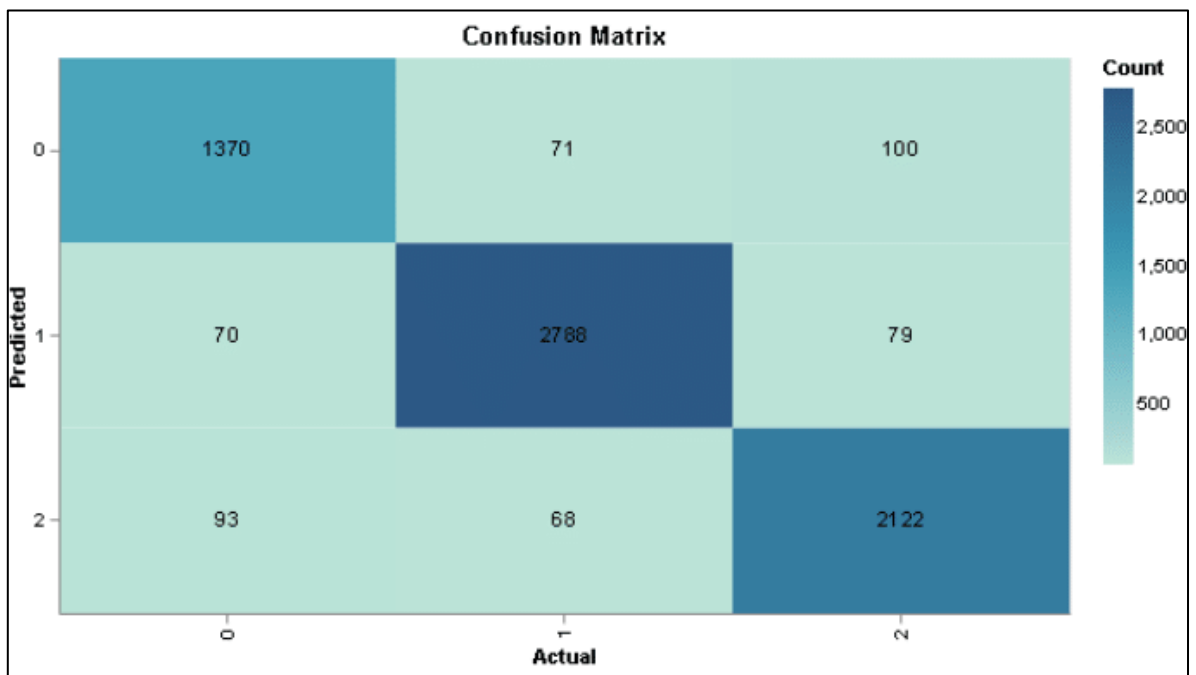


Fig. 4.2.7 Confusion matrix for DistilBERT model

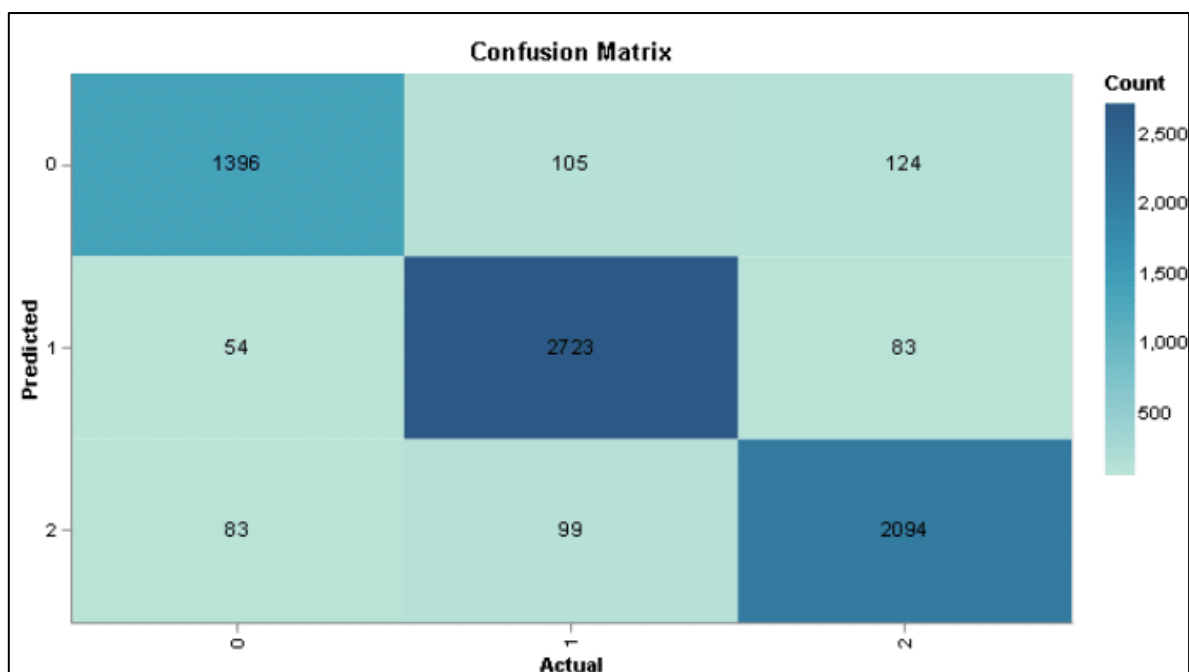


Fig. 4.2.8 Confusion matrix of XLNet model

4.3 CODE AND DATASET SNIPPETS

27020	'Marine uprising has begun': After Suez Canal, a boat blocks Florida highway triggering hilarious jokes online	2
27021	Do it like Vietnam: It maximises thin resources to manage both Covid and the economy superbly	2
27022	Why private sector and consumers must be in sync with a digital economy	1
27023	Donald Trump's attitude towards Pakistan more realistic than recent US Presidents	1
27024	Ben Shapiro - News and Updates on the US Political Commentator	1
27025	Noah Centineo shows off his dramatic physical transformation as he works out shirtless	1
27026	Microsoft Surface Pro X (2020) review: ARM gets more muscle	1
27027	Bernie Sanders against Donald Trump Twitter ban: 'Tomorrow it could be somebody else'	0
27028	Facebook in the dock again: Its different strokes for different folks impair democracy	1
27029	Vivo Nex review	1
27030	Gagged No More: Joe Biden rescinds Mexico City Policy, boosting global women's empowerment movement	0
27031	Debasree Chaudhuri: 'The candidates TMC has fielded, are they good enough to be MLAs?... BJP gives respect to women, not	2
27032	U.S. Open 2020: Put DJ's name on the trophy? Rahm's major breakthrough? - The experts' picks	1
27033	Oil-Price Dynamics	2
27034	Save earth from human nature	2
27035	When America also said #ThankYouRaheelSharif	1
27036	Watch Iran	1
27037	Fortifying milk with Vitamins A and D can improve children's nutrition, boost immunity to infections	2
27038	Communal divide - Who is responsible? (Part 2)	2
27039	Glimpses of grief and resilience, captured over an unforgettable year	0
27040	Suhana Khan's make-up game is on point in new Insta selfie	1
27041	Wind behind their back	1

Snippet of training dataset of news headlines

	A	B
1	Statement	Label
2	Preserving Macao's handmade signs in the digital age	1
3	The Khashoggi mystery: Dismembered body parts of a dissident journalist could be a valuable alliance's unpleasant byproduct	1
4	Let writing rubber cheques stay criminal	0
5	Self-driving vehicles	1
6	Here's how you can set up a successful AI team, IT News, ET CIO	2
7	Fans rally behind Sam Armytage after Kochie and Natalie's Kyle and Jackie O interview	1
8	Asian Games champion Vinesh Phogat tests positive for coronavirus	2
9	Restoring the right to breathe: Migration detention must end	0
10	India isn't Sweden. New York was never.	1
11	Dedicate Padma Shri Award to my team: Hockey captain Rani Rampal	2
12	Confused about the coronavirus? This glossary will help	2
13	Countryside is the new battleground for Nestle India, Britannia and Parle	0
14	Kourtney Kardashian and beau Travis Barker as they attend star-studded UFC 260 event in Las Vegas	1
15	IIT Madras Entrepreneurship Cell to host 'E-Summit'	1
16	Om NaMo SiVa: Silicon Valley values innovation and risk-taking over manufacturing	2
17	Will the private pension age rise from 55 to 57 in April 2028 affect me?	1
18	BJP leader, 2 others booked for flouting Covid-19 rules at marriage reception in Pune	1
19	Taiwan's New Southbound Policy: An initiative to secure the future	2
20	Johnny Damon's arrest video shows the treatment of citizens by police is very much a Black-or-white issue	0
21	EMUI 10 Beta announced for Huawei P30 series, new version brings dark mode and multi-screen support	2
22	Data gouging is just as bad a monopoly pricing	0
23	European Union agrees to impose broad economic sanctions against Russia	0

Snippet of testing dataset of news headlines

```
In [ ]: !pip install transformers
!pip install simpletransformers
!pip install wandb
!pip install nltk
!pip install scikit-learn
!pip install scipy
!pip install beautifulsoup4
import urllib.request,sys,time
from bs4 import BeautifulSoup
from sklearn import softmax
from simpletransformers.classification import ClassificationModel
import pandas as pd
import sklearn
import numpy as np
import pickle
```

Python libraries used

```

import urllib.request,sys,time
from bs4 import BeautifulSoup
import requests
import pandas as pd

pages_to_get= 100

outerframe=[]
for page in range(1,pages_to_get+1):
    print('processing page :', page)
    url = 'https://timesofindia.indiatimes.com/blogs/india/page/'+str(page)
    print(url)

    #an exception might be thrown, so the code should be in a try-except block
    try:
        #use the browser to get the url. This is suspicious command that might blow up.
        page=requests.get(url) # this might throw an exception if something goes wrong.

    except Exception as e: # this describes what to do if an exception is thrown
        error_type, error_obj, error_info = sys.exc_info() # get the exception information
        print ('ERROR FOR LINK:',url) #print the link that cause the problem
        print (error_type, 'Line:', error_info.tb_lineno) #print error info and line that threw the exception
        continue #ignore this page. Abandon this and go back.
    time.sleep(2)
    soup=BeautifulSoup(page.text,'html.parser')
    frame=[]
    links=soup.find_all('div',attrs={'class':'detail'})
    print(len(links))
    for j in links:
        try:
            Statement = j.h2.a.text
        except Exception as e:
            Statement = None
        frame.append((Statement))

    outerframe.extend(frame)
data52=pd.DataFrame(outerframe, columns=['Statement'])
data52

```

Data scraping using beautiful soup library

In []:

```

!pip install newspaper3k
import newspaper

```

```

import pandas as pd
#Create newspaper object
frame1=[]
outer_frame1=[]
bbc=newspaper.build('https://news.yahoo.com/abc-news/')
for article in bbc.articles:
    print(article.url)
    frame1.append(article.url)
outer_frame1.extend(frame1)
data2=pd.DataFrame(outer_frame1, columns=['links'])
data2

```

Data scraping using newspaper library

```
In [ ]: import nltk
nltk.download('vader_lexicon')

[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
[nltk_data] Package vader_lexicon is already up-to-date!
```

Out[]: True

```
In [ ]: from IPython import display
import math
from pprint import pprint
import pandas as pd
import nltk

from nltk.sentiment.vader import SentimentIntensityAnalyzer as SIA
sia = SIA()
final_data1['compound']=[sia.polarity_scores(i) ['compound'] for i in final_data1['Statement']]
final_data1['neg']=[sia.polarity_scores(i) ['neg'] for i in final_data1['Statement']]
final_data1['neu']=[sia.polarity_scores(i) ['neu'] for i in final_data1['Statement']]
final_data1['pos']=[sia.polarity_scores(i) ['pos'] for i in final_data1['Statement']]
```

```
In [ ]: for i in range (len(final_data1)):
    if ((final_data1.loc[i, 'compound'])==0):
        final_data1.loc[i, 'Label']=1
    elif ((final_data1.loc[i, 'compound'] > 0):
        final_data1.loc[i, 'Label']=2
    elif ((final_data1.loc[i, 'compound'] < 0):
        final_data1.loc[i, 'Label']=0
final_data1
```

```
In [ ]: final_data2=final_data1[['Statement', 'Label']]
final_data2
```

Data Labelling process

```
In [ ]: from sklearn.model_selection import train_test_split
train, test= train_test_split(final_data2, test_size=0.20,shuffle=True,random_state=10)
```

Splitting Dataset into training and testing set

```
In [ ]: from simpletransformers.classification import ClassificationModel

model_args = {
    'max_seq_length': 256,
    'num_train_epochs': 5,
    'train_batch_size': 16,
    'eval_batch_size': 32,
    'gradient_accumulation_steps': 1,
    'learning_rate': 3e-5,
    'save_steps': 5000,
    'wandb_project': 'Sentiment5',
    'evaluate_during_training': True,
    'evaluate_during_training_steps': 1000,
    'reprocess_input_data': True,
    'save_model_every_epoch': False,
    'overwrite_output_dir': True,
    'no_cache': True,
    'use_early_stopping': True,
    'early_stopping_patience': 3,
    'manual_seed': 4,
}
model = ClassificationModel("bert", "bert-base-cased", num_labels=3, args=model_args, use_cuda=True)

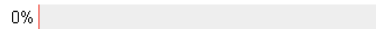
In [ ]: model.train_model(training_data, eval_df=testing_data)
```

Model training and hyperparameter setting for one of the model

```
In [8]: from sklearn.metrics import f1_score, accuracy_score

def f1_multiclass(labels, preds):
    return f1_score(labels, preds, average='micro')

result, model_outputs, wrong_predictions = model.eval_model(testing_data, f1=f1_multiclass, acc=accuracy_score)
```

0% |  14/6761 [00:01<14:52, 7.56it/s]

Running Evaluation: 100% |  212/212 [06:12<00:00, 1.39s/it]

Model evaluation on testing dataset of one of the model

```
In [ ]: pred,output=model.predict(predict_sentence)
print(pred)
```

```
0% | 14/6761 [00:02<19:55, 5.64it/s]
100% | 212/212 [02:41<00:00, 1.68it/s]
```

```
In [ ]: softmax_output=softmax(model_outputs[21])
rounding_softmax_output=[%.6f' % i for i in softmax_output]
print((rounding_softmax_output))
if ((softmax_output[0] > softmax_output[1]) and (softmax_output[0] > softmax_output[2])):
    print ("Negative Sentiment")
elif ((softmax_output[1] > softmax_output[0]) and (softmax_output[1] > softmax_output[2])):
    print ("Neutral Sentiment")
elif ((softmax_output[2] > softmax_output[0]) and (softmax_output[2] > softmax_output[1])):
    print ("Positive Sentiment")
```

```
['0.000425', '0.000247', '0.999329']
Positive Sentiment
```

Prediction and application of softmax function

CONCLUSION

Sentiment analysis is a way of analysing opinions and behaviour from text data; it can provide informative feedback and earnings to a business. According to Table 1 and the visualisations in the results and discussion section, all of the transformer models perform well on the news headline dataset, with the bert-base-cased model outperforming the others. In comparison to papers such as [24] [25], we see that our fine-tuned models outperform both traditional machine learning and deep learning models such as CNN, LSTM, and others. Therefore, transformers are the latest big thing in tasks like text classification, machine translation, language modelling, etc. For future work, the large number of hyperparameters available with simple transformers can be manipulated to allow our models to perform even better as good as human accuracy.

REFERENCES

- [1] Hugging Face, "Transformers," 2020. [Online]. Available: <https://huggingface.co/transformers/>. [Accessed 1 February 2021].
- [2] C. Min, Z. Fanwei, W. Minghui and Y. Jing, "Survey of opinion," *Journal of Zhejiang University (Engineering Science)*, vol. 48, no. 8, pp. 1461-1472, 2014.
- [3] J. Schectman, "Obama's campaign used salesforce.com to gauge feelings," 2012. [Online]. Available: <https://www.wsj.com/articles/BL-CIOB-1295>. [Accessed 30 March 2021].
- [4] D. Dor, "On newspaper headlines as relevance optimizers," *Journal of Pragmatics*, vol. 35, no. 5, pp. 695-721, 2003.
- [5] M. Karlsson, "The immediacy of online news, the visibility of journalistic processes and a restructuring of journalistic authority," *Journalism*, vol. 12, no. 3, pp. 279-295, 2011.
- [6] I. Salián, "SuperVize me: What's the difference between supervised, unsupervised, semisupervised and reinforcement learning?," 2 August 2018. [Online]. Available: <https://blogs.nvidia.com/blog/2018/08/02/supervised-unsupervised-learning/>. [Accessed 25 December 2020].
- [7] T. Wolf *et al.*, "Transformers: State-of-the-art natural language processing," in *Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020.
- [8] T. Rajapakse, "Simple Transformers," 2019. [Online]. Available: <https://www.simpletransformers.ai/>. [Accessed 12 February 2021].

- [9] S. Fong, Y. Zhuang, J. Li and R. Khoury, "Sentiment analysis of online news using MALLET," in *2013 International Symposium on Computational and Business Intelligence (ISCBI)*, 2013.
- [10] J. Reis, F. Benevenuto, P. Olmo, R. Prates, H. Kwak and J. An, "Breaking the news: First impressions matter on online news," in *9th International conference on web and social media (ICWSM)*, 2015.
- [11] N. Godbole, M. Srinivasaiah and S. Skiena, "Large-scale sentiment analysis for news and blogs," in *2007 International conference on weblogs and social media (ICWSM)*, 2007.
- [12] Y. Liu *et al.*, "RoBERTa: A robustly optimized bert pretraining approach," ArXiv, abs/1907.11692, 2019.
- [13] M. Joshi *et al.*, "SpanBERT: Improving pre-training by representing and predicting spans," ArXiv, abs/1907.10529, 2020..
- [14] Z. Yang *et al.*, "Xlnet: Generalized autoregressive pretraining for language understanding," ArXiv, abs/1906.08237, 2019.
- [15] A. Bosselut, H. Rashkin, M. Sap, C. Malaviya, A. Celikyilmaz and Y. Choi, "COMET: Commonsense transformers for automatic knowledge graph construction," in *57th Annual meeting of the association for computational linguistics*, 2019.
- [16] M. Lewis *et al.*, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," ArXiv, abs/1910.13461, 2019..
- [17] G. Lample and A. Conneau, "Cross-lingual language model pretraining," ArXiv, abs/1901.07291, 2019..

- [18] C. Nwankpa, W. Ijomah, A. Gachagan and S. Marshall, "Activation functions: Comparison of trends in practice and research for deep," ArXiv, abs/1811.03378, 2018.
- [19] T. Wood, "Softmax Function," [Online]. Available: <https://deepai.org/machine-learning-glossary-and-terms/softmax-layer>. [Accessed 1 March 2021].
- [20] A. Kumar, "Data Analytics," 16 October 2020. [Online]. Available: <https://vitalflux.com/what-softmax-function-why-needed-machine-learning/>. [Accessed 1 March 2021].
- [21] L. Biewald, "Experiment tracking with weights and biases," 2020. [Online]. Available: <https://www.wandb.com/>. [Accessed 15 March 2021].
- [22] P. Joshi, "Transfer learning for NLP: Fine-Tuning BERT for text classification," 21 July 2020. [Online]. Available: <https://www.analyticsvidhya.com/blog/2020/07/transferlearning-for-nlp-fine-tuning-bert-for-text-classification/>. [Accessed 12 February 2021].
- [23] F. Pedregosa, "Scikit-learn: Machine Learning in {P}ython," *Journal of Machine Learning Research*, vol. 12, pp. 2825--2830, 2011.
- [24] C. J. Rameshbhai and J. Paulose, "Opinion mining on newspaper headlines using SVM and NLP," *International journal of electrical and computer engineering (IJECE)*, vol. 9, no. 3, pp. 2152-2163, 2019.
- [25] S. Jagtap, M. Sonwane, A. Shelke, Y. Bedmuthaont and S. Gore, "Financial news analysis with NLP and machine learning," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 3, no. 4, pp. 623-628, April 2021.