Twitter Sentiment Analysis-RoBERTa model

RoBERTa (Robustly Optimized BERT approach) is a state-of-the-art natural language processing (NLP) model developed by Facebook AI Research. It is based on the Transformer architecture and is an extension of the popular BERT (Bidirectional Encoder Representations from Transformers) model. RoBERTa was trained on a massive amount of text data from the Internet, using a similar methodology as BERT but with several modifications to improve its performance.

RoBERTa has proven to be a powerful model for sentiment analysis and has achieved state-of-the-art performance on various benchmarks and competitions in the field of NLP. Its ability to capture contextual information and understand the nuances of language makes it a valuable tool for sentiment analysis tasks.

Task

Build a model that can rate the sentiment of a Tweet based on its content.

Data

In [1]:

```
import pandas as pd
from sklearn.metrics import accuracy_score
pd.set_option('display.max_colwidth',None)
data=pd.read_csv('tweet_product_company.csv',encoding='unicode_escape')
data.head()
```

Out[1]:

	tweet_text	emotion_in_tweet_is_directed_at	is_there_an_emotion_directed_at_a_brand_
0	.@wesley83 I have a 3G iPhone. After 3 hrs tweeting at #RISE_Austin, it was dead! I need to upgrade. Plugin stations at #SXSW.	iPhone	Negat
1	@jessedee Know about @fludapp ? Awesome iPad/iPhone app that you'll likely appreciate for its design. Also, they're giving free Ts at #SXSW	iPad or iPhone App	Posit
2	@swonderlin Can not wait for #iPad 2 also. They should sale them down at #SXSW.	iPad	Posit
3	@sxsw I hope this year's festival isn't as crashy as this year's iPhone app. #sxsw	iPad or iPhone App	Negat
4	@sxtxstate great stuff on Fri #SXSW: Marissa Mayer (Google), Tim O'Reilly (tech books/conferences) & Matt Mullenweg (Wordpress)	Google	Posit

EDA

In [2]:

```
data.columns
```

Out[2]:

```
Index(['tweet_text', 'emotion_in_tweet_is_directed_at',
       'is_there_an_emotion_directed_at_a_brand_or_product'],
      dtype='object')
```

In [3]:

```
data.shape
```

Out[3]:

```
(9093, 3)
```

In [4]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9093 entries, 0 to 9092
Data columns (total 3 columns):
 #   Column                                              Non-Null Coun
t  Dtype
---  ------                                              --------------
-  -----
 0   tweet_text                                          9092 non-null
object
 1   emotion_in_tweet_is_directed_at                     3291 non-null
object
 2   is_there_an_emotion_directed_at_a_brand_or_product  9093 non-null
object
dtypes: object(3)
memory usage: 213.2+ KB
```

In [5]:

```
data.describe()
```

Out[5]:

	tweet_text	emotion_in_tweet_is_directed_at	is_there_an_emotion_directed_at_a_brand_or
count	9092	3291	
unique	9065	9	
top	RT @mention Marissa Mayer: Google Will Connect the Digital & Physical Worlds Through Mobile - {link} #sxsw	iPad	No emotion toward brand c
freq	5	946	

In [6]:

```
data.isnull().sum()
```

Out[6]:

```
tweet_text                                            1
emotion_in_tweet_is_directed_at                    5802
is_there_an_emotion_directed_at_a_brand_or_product    0
dtype: int64
```

Preprocessing

Dropping Columns

In [7]:

```
data.drop('emotion_in_tweet_is_directed_at',axis=1,inplace=True)
```

Renaming Columns

In [8]:

```
data.rename(columns={
    'tweet_text': 'tweet',
    'is_there_an_emotion_directed_at_a_brand_or_product': 'emotion'
},inplace=True)
data.head()
```

Out[8]:

	tweet	emotion
0	.@wesley83 I have a 3G iPhone. After 3 hrs tweeting at #RISE_Austin, it was dead! I need to upgrade. Plugin stations at #SXSW.	Negative emotion
1	@jessedee Know about @fludapp ? Awesome iPad/iPhone app that you'll likely appreciate for its design. Also, they're giving free Ts at #SXSW	Positive emotion
2	@swonderlin Can not wait for #iPad 2 also. They should sale them down at #SXSW.	Positive emotion
3	@sxsw I hope this year's festival isn't as crashy as this year's iPhone app. #sxsw	Negative emotion
4	@sxtxstate great stuff on Fri #SXSW: Marissa Mayer (Google), Tim O'Reilly (tech books/conferences) & Matt Mullenweg (Wordpress)	Positive emotion

Missing Values

In [9]:

```
data.tweet.fillna('',inplace=True)
data.isnull().sum()
```

Out[9]:

```
tweet      0
emotion    0
dtype: int64
```

Transforming text

In [10]:

```
data.tweet=[tweet.lower() for tweet in data.tweet]
```

Removing Punctuation marks

In [11]:

```
import string
data.tweet = data.tweet.apply(lambda x: x.translate(str.maketrans('', '', string.pu
data[:5]
```

Out[11]:

	tweet	emotion
0	wesley83 i have a 3g iphone after 3 hrs tweeting at riseaustin it was dead i need to upgrade plugin stations at sxsw	Negative emotion
1	jessedee know about fludapp awesome ipadiphone app that youll likely appreciate for its design also theyre giving free ts at sxsw	Positive emotion
2	swonderlin can not wait for ipad 2 also they should sale them down at sxsw	Positive emotion
3	sxsw i hope this years festival isnt as crashy as this years iphone app sxsw	Negative emotion
4	sxtxstate great stuff on fri sxsw marissa mayer google tim oreilly tech booksconferences amp matt mullenweg wordpress	Positive emotion

Encoding Emotions

In [12]:

```
unique_emotions=list(data.emotion.unique())
unique_emotions
```

Out[12]:

```
['Negative emotion',
 'Positive emotion',
 'No emotion toward brand or product',
 "I can't tell"]
```

In [13]:

```
binary_emotions=[]
emotions=data.emotion
for val in emotions:
    if val=='Negative emotion':
        binary_emotions.append(0)
    if val=='Positive emotion':
        binary_emotions.append(1)
    if val=='No emotion toward brand or product':
        binary_emotions.append(2)
    if val=="I can't tell":
        binary_emotions.append(2)
binary_emotions=pd.DataFrame(binary_emotions).rename(columns={0:'binary_emotions'})
data=data.join(binary_emotions)
data.head()
```

Out[13]:

	tweet	emotion	binary_emotions
0	wesley83 i have a 3g iphone after 3 hrs tweeting at riseaustin it was dead i need to upgrade plugin stations at sxsw	Negative emotion	0
1	jessedee know about fludapp awesome ipadiphone app that youll likely appreciate for its design also theyre giving free ts at sxsw	Positive emotion	1
2	swonderlin can not wait for ipad 2 also they should sale them down at sxsw	Positive emotion	1
3	sxsw i hope this years festival isnt as crashy as this years iphone app sxsw	Negative emotion	0
4	sxtxstate great stuff on fri sxsw marissa mayer google tim oreilly tech booksconferences amp matt mullenweg wordpress	Positive emotion	1

Roberta Model

In [28]:

```
# !pip install transformers
from transformers import AutoTokenizer
from transformers import TFAutoModelForSequenceClassification
from scipy.special import softmax
import os
os.environ['CURL_CA_BUNDLE'] = ''
```

In [29]:

```
MODEL=f'cardiffnlp/twitter-roberta-base-sentiment'
tokenizer=AutoTokenizer.from_pretrained(MODEL)
roberta_model=TFAutoModelForSequenceClassification.from_pretrained(MODEL,force_down
```

```
Downloading (…)lve/main/config.json:   0%|          | 0.00/747 [00:00
<?, ?B/s]

Downloading tf_model.h5:   0%|          | 0.00/501M [00:00<?, ?B/s]

All model checkpoint layers were used when initializing TFRobertaForSe
quenceClassification.

All the layers of TFRobertaForSequenceClassification were initialized
from the model checkpoint at cardiffnlp/twitter-roberta-base-sentimen
t.
If your task is similar to the task the model of the checkpoint was tr
ained on, you can already use TFRobertaForSequenceClassification for p
redictions without further training.
```

In [30]:

```
roberta_model.summary()
```

```
Model: "tf_roberta_for_sequence_classification"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 roberta (TFRobertaMainLayer  multiple                 124055040
 )

 classifier (TFRobertaClassi  multiple                 592899
 ficationHead)

=================================================================
Total params: 124,647,939
Trainable params: 124,647,939
Non-trainable params: 0
_________________________________________________________________
```

In [ ]:

```
def roberta_polarity_scores(tweet):
    encoded_tweet=tokenizer(tweet,return_tensors='tf')
    output=roberta_model(**encoded_tweet)
    scores=output[0][0].numpy()
    scores=softmax(scores)
    scores_dict={
        'roberta_neg':scores[0],
        'roberta_neu':scores[1],
        'roberta_pos':scores[2]
    }
    max_score=max(list(scores_dict.values()))
    for k,v in scores_dict.items():
        if v==max_score:
            result=(k)
        else:
            continue;
    if result=='roberta_neg':
        return 0
    if result=='roberta_pos':
        return 1
    if result=='roberta_neu':
        return 2
```

In [27]:

```
roberta_predictions=[]
for i,row in tqdm(data.iterrows(),total=len(data)):
        text=row.tweet
        roberta_predictions.append(roberta_polarity_scores(text))
roberta_predictions[:10]
```

```
  0%|          | 0/9093 [00:00<?, ?it/s]
```

Out[27]:

```
[0, 1, 1, 2, 1, 1, 2, 1, 1, 1]
```

In [28]:

```
accuracy_score(roberta_predictions,binary_emotions)
```

Out[28]:

```
0.659408336082701
```

The RoBERTa model had an accuracy of ~66%