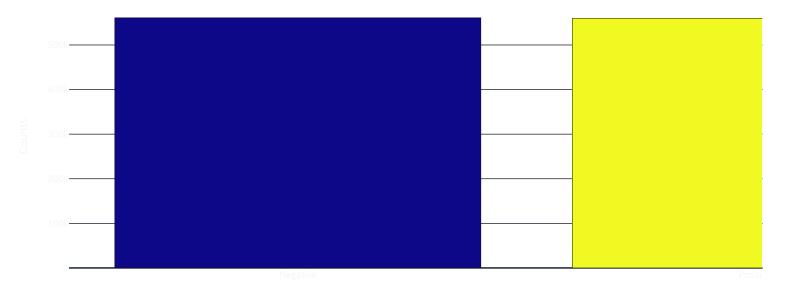
```
import os
import shutil
import tarfile
import tensorflow as tf
from transformers import BertTokenizer, TFBertForSequenceClassification
import pandas as pd
from bs4 import BeautifulSoup
import re
import matplotlib.pyplot as plt
import plotly.express as px
import plotly.offline as pyo
import plotly.graph objects as go
import torch
from wordcloud import WordCloud, STOPWORDS
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification report
from google.colab import drive
drive.mount('/content/gdrive')
    Drive already mounted at /content/gdrive; to attempt to forcibly remount, call drive.mount("/content/gdrive", force remount=True).
tf.test.gpu_device_name()
     '/device:GPU:0'
# Set GPU
device = torch.device("cuda")
from datasets import load_dataset
dataset = load dataset("sentiment140")
train = pd.DataFrame.from_dict(dataset['train'])
train = train.drop(columns=['user', 'date', 'query'])
print(train.head())
                                                     text sentiment
    0 @switchfoot http://twitpic.com/2y1zl - Awww, t...
    1 is upset that he can't update his Facebook by ...
                                                                   0
     2 @Kenichan I dived many times for the ball. Man...
                                                                   0
          my whole body feels itchy and like its on fire
                                                                   0
     4 @nationwideclass no, it's not behaving at all....
```

```
print(train.shape)
     (1600000, 2)
def text_cleaning(text):
    soup = BeautifulSoup(text, "html.parser")
   text = re.sub(r'\[[^]]*\]', '', soup.get_text())
   pattern = r"[^a-zA-Z0-9\s,']"
   text = re.sub(pattern, '', text)
    return text
# Function to convert score to sentiment
def to sentiment(rating):
   rating = int(rating)
    # Convert to class
   if rating < 2:
        return 0
   else:
        return 1
# Apply to the dataset
train['sentiment'] = train.sentiment.apply(to_sentiment)
train_df, test_df = train_test_split(train, test_size=0.3, random_state=42)
sentiment counts = train df['sentiment'].value counts()
fig =px.bar(x= {0:'Negative',1:'Positive'},
           y= sentiment_counts.values,
            color=sentiment counts.index,
            color_discrete_sequence = px.colors.qualitative.Dark24,
            title='<b>Sentiments Counts')
fig.update layout(title='Sentiments Counts',
                xaxis_title='Sentiment',
               yaxis title='Counts',
                template='plotly_dark')
# Show the bar chart
fig.show()
pyo.plot(fig, filename = 'Sentiments Counts.html', auto_open = True)
```

Sentiments Counts



```
# Train dataset
train_df['Cleaned_sentence'] = train_df['text'].apply(text_cleaning).tolist()
# Test dataset
test_df['Cleaned_sentence'] = test_df['text'].apply(text_cleaning).tolist()
```

<ipython-input-10-2374454d2258>:2: MarkupResemblesLocatorWarning:

The input looks more like a filename than markup. You may want to open this file and pass the filehandle into Beautiful Soup.

<ipython-input-10-2374454d2258>:2: MarkupResemblesLocatorWarning:

The input looks more like a filename than markup. You may want to open this file and pass the filehandle into Beautiful Soup.

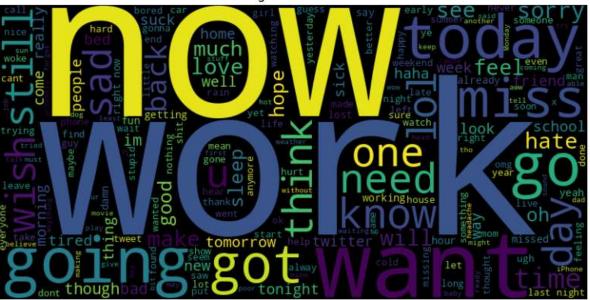
```
# Function to generate word cloud
def generate wordcloud(text,Title):
   all_text = " ".join(text)
   wordcloud = WordCloud(width=800,
                       height=400,
                       stopwords=set(STOPWORDS),
                       background color='black').generate(all text)
   plt.figure(figsize=(10, 5))
   plt.imshow(wordcloud, interpolation='bilinear')
   plt.axis("off")
   plt.title(Title)
   plt.show()
print(train df.head())
                                                          text sentiment \
    1064291 Disnleyland ISN'T tha happiest place in tha wo...
                                                                        1
             OMG BELLLLYYY SHOUTIN AT SUM FOOD.. BRB SOOOO ...
                                                                        0
    281802
    168792
             I'm still waiting to find out what #caca stand...
                                                                        0
    911020
                                               So's your face
     69604
               So sad to see her drive away wat do i do now??
                                              Cleaned sentence
    1064291 Disnleyland ISN'T tha happiest place in tha wo...
    281802
             OMG BELLLLYYY SHOUTIN AT SUM FOOD BRB SOOOO ST...
    168792
             I'm still waiting to find out what caca stands...
    911020
                                               So's your face
     69604
                  So sad to see her drive away wat do i do now
positive = train_df[train_df['sentiment']==1]['Cleaned_sentence'].tolist()
generate_wordcloud(positive, 'Positive Review')
```

Positive Review

tomorrow yeah timesee though 1 Good morning twitter

negative = train_df[train_df['sentiment']==0]['Cleaned_sentence'].tolist()
generate_wordcloud(negative,'Negative Review')

Negative Review



print(test_df.head(30))

	text	sentiment	\
541200	<pre>@chrishasboobs AHHH I HOPE YOUR OK!!!</pre>	0	
750	@misstoriblack cool , i have no tweet apps fo	0	
766711	<pre>@TiannaChaos i know just family drama. its la</pre>	0	
285055	School email won't open and I have geography	0	
705995	upper airways problem	0	
379611	Going to miss Pastor's sermon on Faith	0	
1189018	on lunchdj should come eat with me	1	
667030	<pre>@piginthepoke oh why are you feeling like that?</pre>	0	
93541	gahh noo!peyton needs to live!this is horrible	0	
1097326	@mrstessyman thank you glad you like it! There	1	
1380171	<pre>@PerezHilton Zach makes me pee sitting down! A</pre>	1	
1233933	to sum up my day in one word kackered!	1	
1495501	@k9wkj Great minds think alike	1	
558391	Is Poorly and in bed!	0	
820010	@LilPecan Oh, that's really great Here we hav	1	
986772	@wizely lol, calm down. i got a 30day loan of	1	
1586084	i'm feeling quite sleepy today, wish i could s	1	
209248	@nadalnews Yeah Mathieu totally choked in the	0	
320428	ugh, morning's off to a rough start	0	

```
427095
                                            just bit my tongue
     288475
              @frandrescher Apparently you dont have time fo...
                                                                         0
    1399246
             Whaddup Whaddup Whaddup Whaddup I Got...
                                                                         1
    1042030
              @GogDog I'd be willing to pay $40-50 (thus cov...
                                                                         1
    1589403
             Working on photos from Hillsong's 1 year celeb...
                                                                         1
                                                                         0
    44848
              baby's chicken pox are soo bad on the face, I ...
     625585
                                             i don't feel good
                                                                         0
    1084782
              @edadkins ha ha I want to see pictures from wh...
                                                                         1
    171708
                                                                         0
              This day will pass slowly. All of these days ...
     314466
                                              my car is poorly
                                                                         0
     833693
              im turning 18 one week from now but i don't f...
                                                                         1
                                               Cleaned sentence
                             chrishasboobs AHHH I HOPE YOUR OK
     541200
     750
              misstoriblack cool, i have no tweet apps for...
     766711
             TiannaChaos i know just family drama its lame...
              School email won't open and I have geography ...
     285055
    705995
                                         upper airways problem
     379611
                        Going to miss Pastor's sermon on Faith
    1189018
                            on lunchdj should come eat with me
     667030
                 piginthepoke oh why are you feeling like that
    93541
                  gahh noopeyton needs to livethis is horrible
             mrstessyman thank you glad you like it There i...
    1097326
    1380171
             PerezHilton Zach makes me pee sitting down And...
    1233933
                        to sum up my day in one word kackered
    1495501
                                 k9wkj Great minds think alike
     558391
                                          Is Poorly and in bed
             LilPecan Oh, that's really great Here we have...
    820010
    986772
              wizely lol, calm down i got a 30day loan offe...
    1586084
             i'm feeling quite sleepy today, wish i could s...
     209248
              nadalnews Yeah Mathieu totally choked in the ...
     320428
                           ugh, morning's off to a rough start
     427095
                                           just bit my tongue
     288475
              frandrescher Apparently you don't have time for...
    1399246
             Whaddup Whaddup Whaddup Whaddup I Got...
    1042030
              GogDog I'd be willing to pay 4050 thus coverin...
             Working on photos from Hillsong's 1 year celeb...
    1589403
# Training data
#Reviews = "[CLS] " +train df['Cleaned sentence'] + "[SEP]"
Reviews = train df['Cleaned sentence']
Target = train df['sentiment']
# Test data
#test_reviews = "[CLS] " +test_df['Cleaned_sentence'] + "[SEP]"
test reviews = test df['Cleaned sentence']
test targets = test df['sentiment']
x val, x test, y val, y test = train test split(test reviews,
                                                    test targets,
                                                    test size=0.5,
                                                    stratify = test_targets)
```

```
#Tokenize and encode the data using the BERT tokenizer
tokenizer = BertTokenizer.from pretrained('bert-base-uncased', do lower case=True)
max len= 128
# Tokenize and encode the sentences
X train encoded = tokenizer.batch encode plus(Reviews.tolist(),
                                                                                       padding=True,
                                                                                       truncation=True,
                                                                                       max_length = max_len,
                                                                                       return tensors='tf')
X_val_encoded = tokenizer.batch_encode_plus(x_val.tolist(),
                                                                                       padding=True,
                                                                                       truncation=True,
                                                                                       max_length = max_len,
                                                                                       return tensors='tf')
X_test_encoded = tokenizer.batch_encode_plus(x_test.tolist(),
                                                                                       padding=True,
                                                                                       truncation=True,
                                                                                       max_length = max_len,
                                                                                       return tensors='tf')
k = 0
print('Training Comments -->>',Reviews[k])
print('\nInput Ids -->>\n',X_train_encoded['input_ids'][k])
print('\nDecoded Ids -->>\n',tokenizer.decode(X train encoded['input ids'][k]))
print('\nAttention Mask -->>\n',X_train_encoded['attention_mask'][k])
print('\nLabels -->>',Target[k])
         Training Comments -->> switchfoot httptwitpiccom2y1zl Awww, that's a bummer You shoulda got David Carr of Third Day to do it D
          Input Ids -->>
            tf.Tensor(
           [ 101 4487 2015 20554 3240 3122 3475 1005 1056 22794 5292 9397
            10458 2173 1999 22794 2088 24665 4215 4440 2064 11891 2026 10338
              1010 2092 2672 2025 3894 3137 2049 1037 2235 2088 4536
                                                                                                                                                   102
                                                                    0
                                                                                            0
                                                                                                                    0
                                                                                                                                0
                                                                                                                                           0
                                                                                                                                                        0
                    0
                                0
                                            0
                                                        0
                                                                                0
                                                                                                        0
                    0
                                                                    0
                                                                                0
                                                                                            0
                                                                                                                                0
                                                                                                                                            0
                    0
                                0
                                                                                0
                                                                                                        0
                                                                                                                                0
                                                                                                                                           0
                                                                                                                                                        0
                    0
                                                        0], shape=(76,), dtype=int32)
         Decoded Ids -->>
            [CLS] disnleyland isn't tha happiest place in tha world grad trip can suck my cock, well maybe not magic mountain its a small world ride [SEP] [PAD] [
         Attention Mask -->>
            tf.Tensor(
          0 0], shape=(76,), dtype=int32)
```

```
# Intialize the model
model = TFBertForSequenceClassification.from pretrained('bert-base-uncased', num labels=2)
    model.safetensors: 100%
                                                        440M/440M [00:02<00:00, 197MB/s]
    All PyTorch model weights were used when initializing TFBertForSequenceClassification.
    Some weights or buffers of the TF 2.0 model TFBertForSequenceClassification were not initialized from the PyTorch model and are r
    You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
# Compile the model with an appropriate optimizer, loss function, and metrics
optimizer = tf.keras.optimizers.Adam(learning_rate=2e-5)
loss = tf.keras.losses.SparseCategoricalCrossentropy(from logits=True)
metric = tf.keras.metrics.SparseCategoricalAccuracy('accuracy')
model.compile(optimizer=optimizer, loss=loss, metrics=[metric])
# Step 5: Train the model
history = model.fit(
   [X train encoded['input ids'], X train encoded['token type ids'], X train encoded['attention mask']],
   Target,
   validation_data=(
   [X val encoded['input ids'], X val encoded['token type ids'], X val encoded['attention mask']], y val),
   batch size=32,
   epochs=3
    Epoch 1/3
    35000/35000 [============= ] - 18874s 538ms/step - loss: 0.3501 - accuracy: 0.8456 - val loss: 0.3230 - val accuracy: 0.8593
    Epoch 2/3
    Epoch 3/3
    35000/35000 [============ - 18798s 537ms/step - loss: 0.2388 - accuracy: 0.9017 - val loss: 0.3402 - val accuracy: 0.8602
#Evaluate the model on the test data
test_loss, test_accuracy = model.evaluate(
   [X test encoded['input ids'], X test encoded['token type ids'], X test encoded['attention mask']],
   y_test
print(f'Test loss: {test_loss}, Test accuracy: {test_accuracy}')
    Test loss: 0.34104079008102417, Test accuracy: 0.8593666553497314
```

```
model_save_name = 'Model_TweetSentimentClassifier'
path = F"/content/gdrive/MyDrive/Colab Notebooks/NLP_ClassProject"
model.save_weights(os.path.join(path, model_save_name))
print(model.summary)
```

<bound method Model.summary of <transformers.models.bert.modeling_tf_bert.TFBertForSequenceClassification object at 0x787af00b58a0>>