

```
In [418... import nltk
nltk.download('punkt')
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
from nltk import stem
stemmer = stem.PorterStemmer()
from nltk import word_tokenize
nltk.download('stopwords')
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
import string
punct = list(string.punctuation)
punctuations = string.punctuation
from collections import Counter
import requests
import pandas as pd
import seaborn as sns
sns.set()
import matplotlib.pyplot as plt
!pip install PRAW
import numpy as np
import praw
import datetime
import time
import os
!pip install plotly
import plotly.express as px
!pip install jupyterlab "ipywidgets>=7.5"
from nltk.corpus import wordnet as wn
import plotly.graph_objects as go
import csv
vad = pd.read_excel('vad.xlsx', index_col = 0)
```

```
[nltk_data] Downloading package punkt to /home/jovyan/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /home/jovyan/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Requirement already satisfied: PRAW in /opt/conda/lib/python3.10/site-packages (7.6.1)  
Requirement already satisfied: websocket-client<=0.54.0 in /opt/conda/lib/python3.10/site-packages (from PRAW) (1.5.1)  
Requirement already satisfied: prawcore<3,>=2.1 in /opt/conda/lib/python3.10/site-packages (from PRAW) (2.3.0)  
Requirement already satisfied: update-checker<=0.18 in /opt/conda/lib/python3.10/site-packages (from PRAW) (0.18.0)  
Requirement already satisfied: requests<3.0,>=2.6.0 in /opt/conda/lib/python3.10/site-packages (from prawcore<3,>=2.1->PRAW) (2.28.2)  
Requirement already satisfied: certifi<=2017.4.17 in /opt/conda/lib/python3.10/site-packages (from requests<3.0,>=2.6.0->prawcore<3,>=2.1->PRAW) (2022.12.7)  
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/conda/lib/python3.10/site-packages (from requests<3.0,>=2.6.0->prawcore<3,>=2.1->PRAW) (1.26.14)  
Requirement already satisfied: charset-normalizer<4,>=2 in /opt/conda/lib/python3.10/site-packages (from requests<3.0,>=2.6.0->prawcore<3,>=2.1->PRAW) (2.1.1)  
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/lib/python3.10/site-packages (from requests<3.0,>=2.6.0->prawcore<3,>=2.1->PRAW) (2.8)  
Requirement already satisfied: plotly in /opt/conda/lib/python3.10/site-packages (5.13.0)  
Requirement already satisfied: tenacity<=6.2.0 in /opt/conda/lib/python3.10/site-packages (from plotly) (8.2.1)  
Requirement already satisfied: jupyterlab in /opt/conda/lib/python3.10/site-packages (3.6.1)  
Requirement already satisfied: ipywidgets<=7.5 in /opt/conda/lib/python3.10/site-packages (8.0.4)  
Requirement already satisfied: nbclassic in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (0.5.1)  
Requirement already satisfied: tornado<=6.1.0 in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (6.2)  
Requirement already satisfied: jinja2<=2.1 in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (3.1.2)  
Requirement already satisfied: packaging in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (23.0)  
Requirement already satisfied: jupyter-server-ydoc<0.7.0,>=0.6.0 in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (0.6.1)  
Requirement already satisfied: tomli in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (2.0.1)  
Requirement already satisfied: notebook<7 in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (6.5.2)  
Requirement already satisfied: jupyter-core in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (5.2.0)  
Requirement already satisfied: jupyter-server<3,>=1.16.0 in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (1.23.6)  
Requirement already satisfied: ipython in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (8.10.0)  
Requirement already satisfied: jupyter-ydoc<=0.2.2 in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (0.2.2)  
Requirement already satisfied: jupyterlab-server<=2.19 in /opt/conda/lib/python3.10/site-packages (from jupyterlab) (2.19.0)  
Requirement already satisfied: traitlets<=4.3.1 in /opt/conda/lib/python3.10/site-packages (from ipywidgets<=7.5) (5.9.0)  
Requirement already satisfied: jupyterlab-widgets<=3.0 in /opt/conda/lib/python3.10/site-packages (from ipywidgets<=7.5) (3.0.5)  
Requirement already satisfied: widgetsnbextension<=4.0 in /opt/conda/lib/python3.10/site-packages (from ipywidgets)

>=7.5) (4.0.5)  
Requirement already satisfied: ipykernel>=4.5.1 in /opt/conda/lib/python3.10/site-packages (from ipywidgets>=7.5) (6.21.2)  
Requirement already satisfied: jupyter-client>=6.1.12 in /opt/conda/lib/python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets>=7.5) (8.0.3)  
Requirement already satisfied: pyzmq>=20 in /opt/conda/lib/python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets>=7.5) (25.0.0)  
Requirement already satisfied: comm>=0.1.1 in /opt/conda/lib/python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets>=7.5) (0.1.2)  
Requirement already satisfied: matplotlib-inline>=0.1 in /opt/conda/lib/python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets>=7.5) (0.1.6)  
Requirement already satisfied: nest-asyncio in /opt/conda/lib/python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets>=7.5) (1.5.6)  
Requirement already satisfied: psutil in /opt/conda/lib/python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets>=7.5) (5.9.4)  
Requirement already satisfied: debugpy>=1.6.5 in /opt/conda/lib/python3.10/site-packages (from ipykernel>=4.5.1->ipywidgets>=7.5) (1.6.6)  
Requirement already satisfied: pickleshare in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (0.7.5)  
Requirement already satisfied: pygments>=2.4.0 in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (2.14.0)  
Requirement already satisfied: prompt-toolkit<3.1.0,>=3.0.30 in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (3.0.36)  
Requirement already satisfied: backcall in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (0.2.0)  
Requirement already satisfied: pexpect>4.3 in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (4.8.0)  
Requirement already satisfied: stack-data in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (0.6.2)  
Requirement already satisfied: decorator in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (5.1.1)  
Requirement already satisfied: jedi>=0.16 in /opt/conda/lib/python3.10/site-packages (from ipython->jupyterlab) (0.18.2)  
Requirement already satisfied: MarkupSafe>=2.0 in /opt/conda/lib/python3.10/site-packages (from jinja2>=2.1->jupyterlab) (2.1.2)  
Requirement already satisfied: platformdirs>=2.5 in /opt/conda/lib/python3.10/site-packages (from jupyter-core->jupyterlab) (3.0.0)  
Requirement already satisfied: argon2-cffi in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3,>=1.16.0->jupyterlab) (21.3.0)  
Requirement already satisfied: terminado>=0.8.3 in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3,>=1.16.0->jupyterlab) (0.17.1)

Requirement already satisfied: nbconvert>=6.4.4 in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3, >=1.16.0->jupyterlab) (7.2.9)

Requirement already satisfied: Send2Trash in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3, >=1.16.0->jupyterlab) (1.8.0)

Requirement already satisfied: anyio<4, >=3.1.0 in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3, >=1.16.0->jupyterlab) (3.6.2)

Requirement already satisfied: nbformat>=5.2.0 in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3, >=1.16.0->jupyterlab) (5.7.3)

Requirement already satisfied: prometheus-client in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3, >=1.16.0->jupyterlab) (0.16.0)

Requirement already satisfied: websocket-client in /opt/conda/lib/python3.10/site-packages (from jupyter-server<3, >=1.16.0->jupyterlab) (1.5.1)

Requirement already satisfied: jupyter-server-fileid<1, >=0.6.0 in /opt/conda/lib/python3.10/site-packages (from jupyter-server-ydoc<0.7.0, >=0.6.0->jupyterlab) (0.7.0)

Requirement already satisfied: ypy-websocket<0.9.0, >=0.8.2 in /opt/conda/lib/python3.10/site-packages (from jupyter-server-ydoc<0.7.0, >=0.6.0->jupyterlab) (0.8.2)

Requirement already satisfied: y-py<0.6.0, >=0.5.3 in /opt/conda/lib/python3.10/site-packages (from jupyter-ydoc<0.2.2->jupyterlab) (0.5.5)

Requirement already satisfied: json5>=0.9.0 in /opt/conda/lib/python3.10/site-packages (from jupyterlab-server<2.19->jupyterlab) (0.9.5)

Requirement already satisfied: requests>=2.28 in /opt/conda/lib/python3.10/site-packages (from jupyterlab-server<2.19->jupyterlab) (2.28.2)

Requirement already satisfied: jsonschema>=4.17.3 in /opt/conda/lib/python3.10/site-packages (from jupyterlab-server<2.19->jupyterlab) (4.17.3)

Requirement already satisfied: babel>=2.10 in /opt/conda/lib/python3.10/site-packages (from jupyterlab-server<2.19->jupyterlab) (2.11.0)

Requirement already satisfied: ipython-genutils in /opt/conda/lib/python3.10/site-packages (from notebook<7->jupyterlab) (0.2.0)

Requirement already satisfied: notebook-shim>=0.1.0 in /opt/conda/lib/python3.10/site-packages (from nbclassic->jupyterlab) (0.2.2)

Requirement already satisfied: idna>=2.8 in /opt/conda/lib/python3.10/site-packages (from anyio<4, >=3.1.0->jupyter-server<3, >=1.16.0->jupyterlab) (2.8)

Requirement already satisfied: sniffio>=1.1 in /opt/conda/lib/python3.10/site-packages (from anyio<4, >=3.1.0->jupyter-server<3, >=1.16.0->jupyterlab) (1.3.0)

Requirement already satisfied: pytz>=2015.7 in /opt/conda/lib/python3.10/site-packages (from babel>=2.10->jupyterlab-server<2.19->jupyterlab) (2022.7.1)

Requirement already satisfied: parso<0.9.0, >=0.8.0 in /opt/conda/lib/python3.10/site-packages (from jedi>=0.16->ipython->jupyterlab) (0.8.3)

Requirement already satisfied: pyparsing!=0.17.0, !=0.17.1, !=0.17.2, >=0.14.0 in /opt/conda/lib/python3.10/site-packages (from jsonschema>=4.17.3->jupyterlab-server<2.19->jupyterlab) (0.19.3)

Requirement already satisfied: attrs>=17.4.0 in /opt/conda/lib/python3.10/site-packages (from jsonschema>=4.17.3->

jupyterlab-server~=2.19->jupyterlab) (19.3.0)  
Requirement already satisfied: python-dateutil>=2.8.2 in /opt/conda/lib/python3.10/site-packages (from jupyter-client>=6.1.12->ipykernel>=4.5.1->ipywidgets>=7.5) (2.8.2)  
Requirement already satisfied: jupyter-events>=0.5.0 in /opt/conda/lib/python3.10/site-packages (from jupyter-server-fileid<1,>=0.6.0->jupyter-server-ydoc<0.7.0,>=0.6.0->jupyterlab) (0.6.3)  
Requirement already satisfied: bleach in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (6.0.0)  
Requirement already satisfied: jupyterlab-pygments in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (0.2.2)  
Requirement already satisfied: nbclient>=0.5.0 in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (0.7.2)  
Requirement already satisfied: mistune<3,>=2.0.3 in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (2.0.5)  
Requirement already satisfied: defusedxml in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (0.7.1)  
Requirement already satisfied: tinycss2 in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (1.2.1)  
Requirement already satisfied: beautifulsoup4 in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (4.11.2)  
Requirement already satisfied: pandocfilters>=1.4.1 in /opt/conda/lib/python3.10/site-packages (from nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (1.5.0)  
Requirement already satisfied: fastjsonschema in /opt/conda/lib/python3.10/site-packages (from nbformat>=5.2.0->jupyter-server<3,>=1.16.0->jupyterlab) (2.16.2)  
Requirement already satisfied: ptyprocess>=0.5 in /opt/conda/lib/python3.10/site-packages (from pexpect>4.3->ipython->jupyterlab) (0.7.0)  
Requirement already satisfied: wcwidth in /opt/conda/lib/python3.10/site-packages (from prompt-toolkit<3.1.0,>=3.0.30->ipython->jupyterlab) (0.2.6)  
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/conda/lib/python3.10/site-packages (from requests>=2.28->jupyterlab-server~=2.19->jupyterlab) (1.26.14)  
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/lib/python3.10/site-packages (from requests>=2.28->jupyterlab-server~=2.19->jupyterlab) (2022.12.7)  
Requirement already satisfied: charset-normalizer<4,>=2 in /opt/conda/lib/python3.10/site-packages (from requests>=2.28->jupyterlab-server~=2.19->jupyterlab) (2.1.1)  
Requirement already satisfied: aiosqlite<1,>=0.17.0 in /opt/conda/lib/python3.10/site-packages (from ypy-websocket<0.9.0,>=0.8.2->jupyter-server-ydoc<0.7.0,>=0.6.0->jupyterlab) (0.18.0)  
Requirement already satisfied: aiofiles<23,>=22.1.0 in /opt/conda/lib/python3.10/site-packages (from ypy-websocket<0.9.0,>=0.8.2->jupyter-server-ydoc<0.7.0,>=0.6.0->jupyterlab) (22.1.0)  
Requirement already satisfied: argon2-cffi-bindings in /opt/conda/lib/python3.10/site-packages (from argon2-cffi->jupyter-server<3,>=1.16.0->jupyterlab) (21.2.0)  
Requirement already satisfied: pure-eval in /opt/conda/lib/python3.10/site-packages (from stack-data->ipython->jupyterlab) (0.2.2)

Requirement already satisfied: executing>=1.2.0 in /opt/conda/lib/python3.10/site-packages (from stack-data->ipython->jupyterlab) (1.2.0)

Requirement already satisfied: asttokens>=2.1.0 in /opt/conda/lib/python3.10/site-packages (from stack-data->ipython->jupyterlab) (2.2.1)

Requirement already satisfied: six in /opt/conda/lib/python3.10/site-packages (from asttokens>=2.1.0->stack-data->ipython->jupyterlab) (1.16.0)

Requirement already satisfied: pyyaml>=5.3 in /opt/conda/lib/python3.10/site-packages (from jupyter-events>=0.5.0->jupyter-server-fileid<1,>=0.6.0->jupyter-server-ydoc<0.7.0,>=0.6.0->jupyterlab) (6.0)

Requirement already satisfied: rfc3986-validator>=0.1.1 in /opt/conda/lib/python3.10/site-packages (from jupyter-events>=0.5.0->jupyter-server-fileid<1,>=0.6.0->jupyter-server-ydoc<0.7.0,>=0.6.0->jupyterlab) (0.1.1)

Requirement already satisfied: rfc3339-validator in /opt/conda/lib/python3.10/site-packages (from jupyter-events>=0.5.0->jupyter-server-fileid<1,>=0.6.0->jupyter-server-ydoc<0.7.0,>=0.6.0->jupyterlab) (0.1.4)

Requirement already satisfied: python-json-logger>=2.0.4 in /opt/conda/lib/python3.10/site-packages (from jupyter-events>=0.5.0->jupyter-server-fileid<1,>=0.6.0->jupyter-server-ydoc<0.7.0,>=0.6.0->jupyterlab) (2.0.6)

Requirement already satisfied: cffi>=1.0.1 in /opt/conda/lib/python3.10/site-packages (from argon2-cffi-bindings->argon2-cffi->jupyter-server<3,>=1.16.0->jupyterlab) (1.15.1)

Requirement already satisfied: soupsieve>1.2 in /opt/conda/lib/python3.10/site-packages (from beautifulsoup4->nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (2.3.2.post1)

Requirement already satisfied: webencodings in /opt/conda/lib/python3.10/site-packages (from bleach->nbconvert>=6.4.4->jupyter-server<3,>=1.16.0->jupyterlab) (0.5.1)

Requirement already satisfied: pyparser in /opt/conda/lib/python3.10/site-packages (from cffi>=1.0.1->argon2-cffi-bindings->argon2-cffi->jupyter-server<3,>=1.16.0->jupyterlab) (2.21)

Requirement already satisfied: webcolors>=1.11 in /opt/conda/lib/python3.10/site-packages (from jsonschema>=4.17.3->jupyterlab-server~2.19->jupyterlab) (1.12)

Requirement already satisfied: uri-template in /opt/conda/lib/python3.10/site-packages (from jsonschema>=4.17.3->jupyterlab-server~2.19->jupyterlab) (1.2.0)

Requirement already satisfied: fqdn in /opt/conda/lib/python3.10/site-packages (from jsonschema>=4.17.3->jupyterlab-server~2.19->jupyterlab) (1.5.1)

Requirement already satisfied: jsonpointer>1.13 in /opt/conda/lib/python3.10/site-packages (from jsonschema>=4.17.3->jupyterlab-server~2.19->jupyterlab) (2.3)

Requirement already satisfied: isoduration in /opt/conda/lib/python3.10/site-packages (from jsonschema>=4.17.3->jupyterlab-server~2.19->jupyterlab) (20.11.0)

Requirement already satisfied: arrow>=0.15.0 in /opt/conda/lib/python3.10/site-packages (from isoduration->jsonschema>=4.17.3->jupyterlab-server~2.19->jupyterlab) (1.2.3)

In [419...

#DM = The Daily Mail

#TG = The Guardian

#API KEY IS CONFIDENTIAL

#MY ENQUIRY IS LIMITED TO 50 VIDEOS PER PUBLICATION DUE TO AVAILABLE API

```

In [420... #Admin for YouTube API
API_KEY = "AIzaSyANsLb9I3d9sMCNq9Vy2R9dPKWqcuNmjc8"
DM_CHANNEL_ID = "UCrFXs-7T3dCTv1aYmdArzFQ"
TG_CHANNEL_ID = "UCIRYBXDze5krPDzAE0xFGVA"

In [421... #This project aims to compare the language used in the titles of videos (published on YouTube) by TG and DM – throu
#I have a null hypothesis that the DM scores higher in all VAD metrics, as I believe it is more sensationalist than

In [422... #SECTION 1: DAILY MAIL ANALYSIS.

In [423... #Getting/parsing Daily Mail's YouTube stats. Sorted by descending view count.
DM_url = "https://www.googleapis.com/youtube/v3/search?key="+API_KEY+"&channelId="+DM_CHANNEL_ID+"&part=snippet,id&
DM_response = requests.get(DM_url).json()

In [424... #Obtaining Daily Mail's YouTube titles, joined together into long list.
DM_title_list = []
for video in DM_response['items']:
    if video['id']['kind'] == "youtube#video":
        DM_titles = video['snippet']['title']
        #Titles needed amending due to error with display of character '
        DM_titles = str(DM_titles).replace("&#39;", "\'")
        DM_title_list.append(DM_titles)

In [425... #Translating list of titles into string
DM_title_long_string = " ".join(DM_title_list)

#Cleaning up
DM_title_string = DM_title_long_string.replace('Jungle Confidential', '')
DM_title_string_complete = DM_title_string.replace('|', '')

#Record amount of punctuation in text
DM_num_punctuation = len([token for token in DM_title_string_complete if token in punctuations])

#Record amount of UPPER CASE TEXT
DM_num_uppercase = len([word for word in DM_title_string_complete if word.isupper()])
DM_PROP_UPPER = DM_num_uppercase / len(DM_title_string_complete)

#Tokenise words

```

```

DM_tokenised_words = word_tokenize(DM_title_string_complete)

#lowercase
DM_lc_tokens = [i.lower() for i in DM_tokenised_words]

#Remove punctuation
DM_lc_filtered_tokens = [word for word in DM_lc_tokens if word not in punctuations]

#Remove stop words
DM_lc_fully_filtered_tokens = [word for word in DM_lc_filtered_tokens if not word in stop_words]

#Lemmatize into tokens
DM_lemmatized_tokens = []
for token in DM_lc_fully_filtered_tokens:
    DM_lemmatized_tokens.append(lemmatizer.lemmatize(token))

```

```

In [426... #Word counts
DM_word_counts = pd.Series(Counter(DM_lemmatized_tokens))
DM_word_counts.sort_values()

#DF of counts
DM_df = pd.DataFrame()
DM_df['DM_Lemma_Counts'] = DM_word_counts

#CSV file for your convenience
DM_df.to_csv('DM_Lemma_Counts.csv', index=True)

```

```

In [427... #Gather VAD data
DM_words = []
DM_emo = []

for i in DM_lemmatized_tokens:
    if i in vad.index:
        DM_emo.append(vad.loc[i])
        DM_words.append(i)
    else:
        pass

#DF charting VAD scores
DM_emo_df = pd.DataFrame(DM_emo, index = DM_words)

```



```
#.CSV file for your convenience
DM_emo_df.to_csv('DM_VAD_Counts.csv', index=True)
```

```
In [428... #Calculating means of VAD scores
DM_avg_v = DM_emo_df['valence'].mean()
DM_avg_a = DM_emo_df['arousal'].mean()
DM_avg_d = DM_emo_df['dominance'].mean()
```

```
In [429... #SECTION 2: THE GUARDIAN ANALYSIS.
```

```
In [430... #Getting/parsing The Guardian's YouTube stats. Sorted by descending view count.
TG_url = "https://www.googleapis.com/youtube/v3/search?key="+API_KEY+"&channelId="+TG_CHANNEL_ID+"&part=snippet,id&maxResults=50"
TG_response = requests.get(TG_url).json()
```

```
In [431... #Obtaining The Guardian's YouTube titles, joined together into long list.
TG_title_list = []
for video in TG_response['items']:
    if video['id']['kind'] == "youtube#video":
        TG_titles = video['snippet']['title']
        #Titles needed amending due to error with display of character '
        TG_titles = str(TG_titles).replace("&#39;", "\'")
        TG_title_list.append(TG_titles)
```

```
In [432... #Converting list to .csv for your convenience.
with open('TG_YouTube_Titles.csv', 'w', newline='') as csvfile:
    writer = csv.writer(csvfile)
    writer.writerows(TG_title_list)
```

```
In [433... #Translating list of titles into string
TG_title_long_string = " ".join(TG_title_list)

#Record amount of punctuation in text
TG_num_punctuation = len([token for token in TG_title_long_string if token in punctuations])

#Record amount of UPPER CASE TEXT
TG_num_uppercase = len([word for word in TG_title_long_string if word.isupper()])
TG_PROP_UPPER = TG_num_uppercase / len(TG_title_long_string)
```

```

#Tokenise words
TG_tokenised_words = word_tokenize(TG_title_long_string)

#lowercase
TG_lc_tokens = [i.lower() for i in TG_tokenised_words]

#Remove punctuation
TG_lc_filtered_tokens = [word for word in TG_lc_tokens if word not in punctuations]

#Remove stop words
TG_lc_fully_filtered_tokens = [word for word in TG_lc_filtered_tokens if not word in stop_words]

#Lemmatize into tokens
TG_lemmatized_tokens = []
for token in TG_lc_fully_filtered_tokens:
    TG_lemmatized_tokens.append(lemmatizer.lemmatize(token))

```

In [434...

```

#Word counts
TG_word_counts = pd.Series(Counter(TG_lemmatized_tokens))
TG_word_counts.sort_values()

#DF of counts
TG_df = pd.DataFrame()
TG_df['DM_Lemma_Counts'] = TG_word_counts

#.CSV file for your convenience
TG_df.to_csv('TG_Lemma_Counts.csv', index=True)

```

In [435...

```

#Gather VAD data
TG_words = []
TG_emo = []

for i in TG_lemmatized_tokens:
    if i in vad.index:
        TG_emo.append(vad.loc[i])
        TG_words.append(i)
    else:
        pass

#DF charting VAD scores

```

```
TG_emo_df = pd.DataFrame(TG_emo, index = TG_words)

#.CSV file for your convenience
TG_emo_df.to_csv('TG_VAD_Counts.csv', index=True)
```

```
In [436... #Calculating means of VAD scores
TG_avg_v = TG_emo_df['valence'].mean()
TG_avg_a = TG_emo_df['arousal'].mean()
TG_avg_d = TG_emo_df['dominance'].mean()
```

```
In [437... #SECTION 3: ANALYSIS BETWEEN THE GUARDIAN AND DAILY MAIL.
```

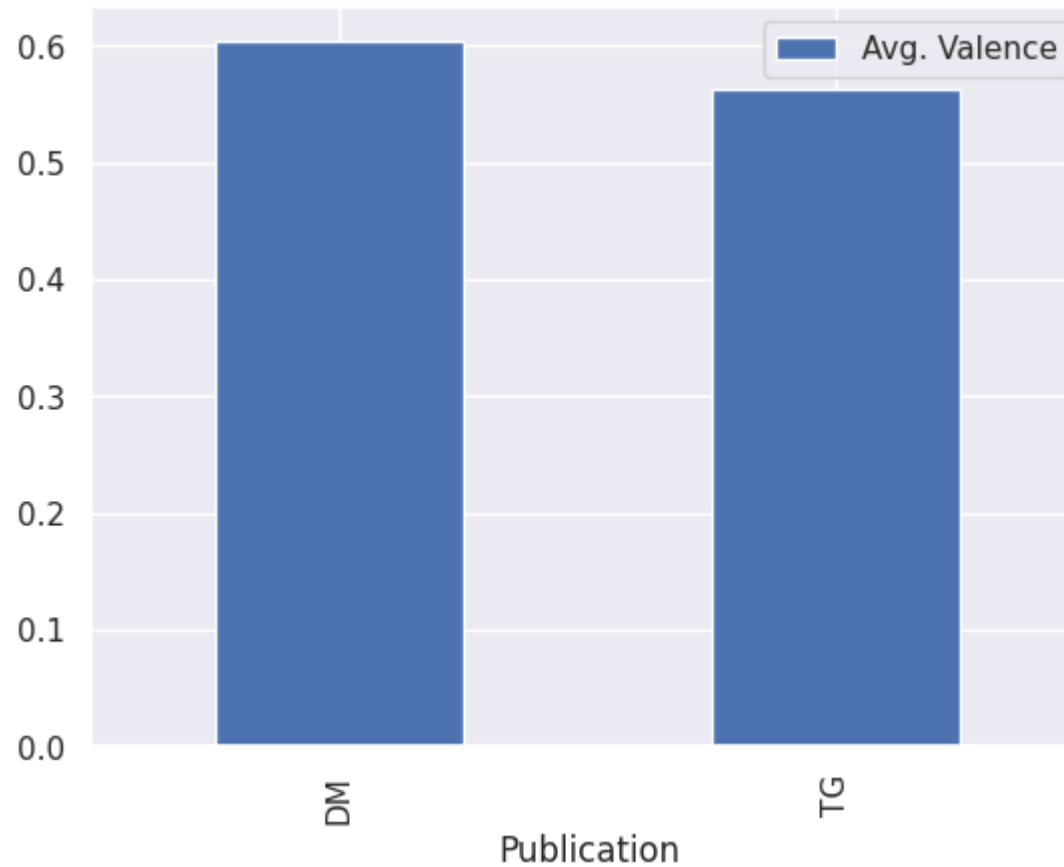
```
In [438... #Comparing means of VAD scores
Analysis_data = {'Publication': ['DM', 'TG'],
                 'Avg. Valence': [DM_avg_v, TG_avg_v],
                 'Avg. Arousal': [DM_avg_a, TG_avg_a],
                 'Avg. Dominance': [DM_avg_d, TG_avg_d]}
comp_df = pd.DataFrame(Analysis_data)

#.CSV file for your convenience
comp_df.to_csv('Average_VAD_Counts_Analysis.csv', index=True)
```

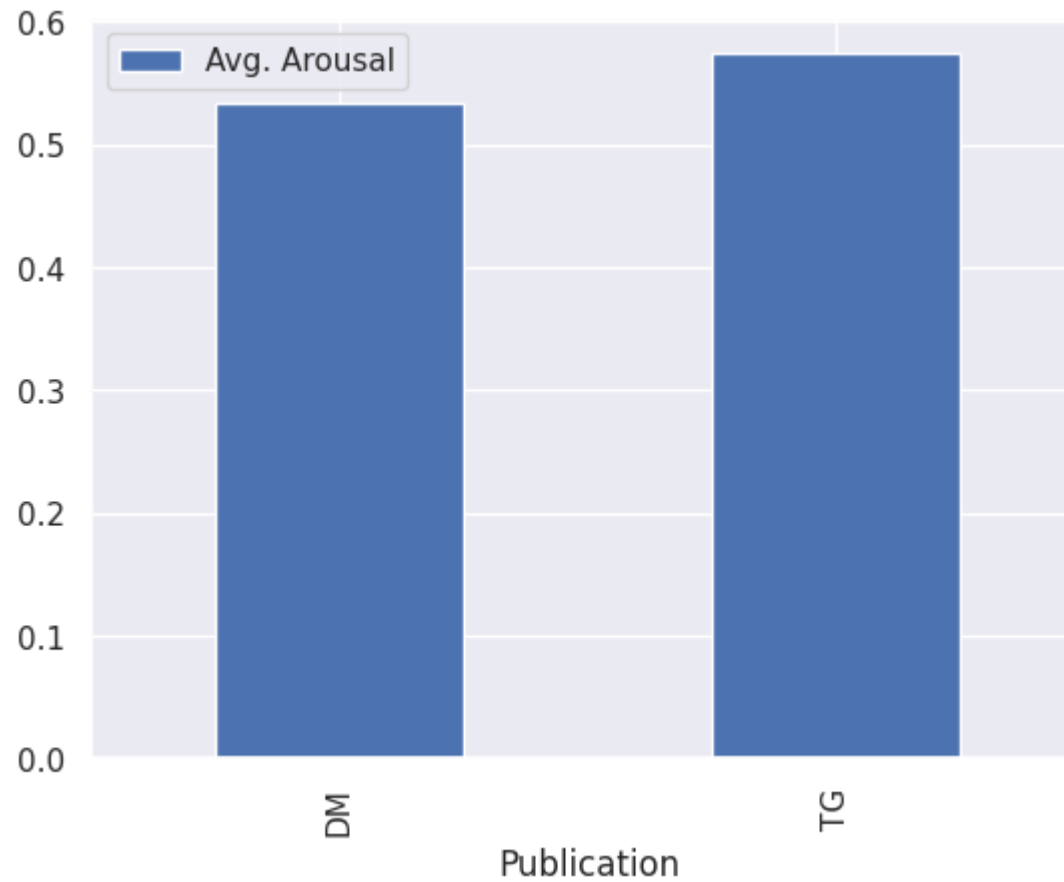
```
In [439... #Plotting average VAD against publications.
comp_df.plot(x='Publication', y='Avg. Valence', kind='bar')

#The Daily Mail uses happier language than The Guardian. DM could be seen to appeal to emotions more than TG.
```

```
Out[439]: <Axes: xlabel='Publication'>
```

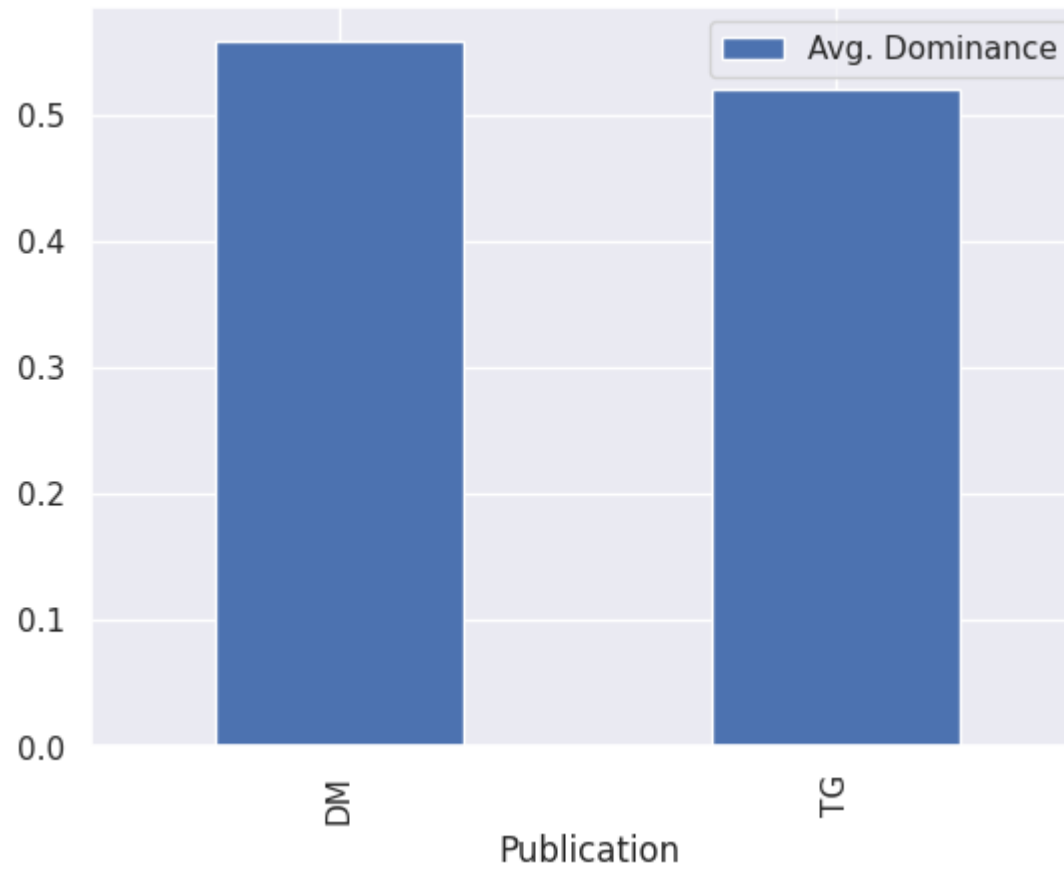


```
In [440]: comp_df.plot(x='Publication', y='Avg. Arousal', kind='bar')  
  
#The Guardian uses more arousing language than the Daily Mail.  
#This suprised me as I expected the DM to score consciderably higher than TG in arousal, as my null hypothesis assu  
  
Out[440]: <Axes: xlabel='Publication'>
```



```
In [441]: comp_df.plot(x='Publication', y='Avg. Dominance', kind='bar')  
#The Daily Mail uses more dominant language than The Guardian. This infers that DM uses a more directive tone.
```

```
Out[441]: <Axes: xlabel='Publication'>
```



In [442... *#Plotting TG and DM VAD values in 3D.*

```
x1 = DM_emo_df['valence']
y1 = DM_emo_df['arousal']
z1 = DM_emo_df['dominance']

x2 = TG_emo_df['valence']
y2 = TG_emo_df['arousal']
z2 = TG_emo_df['dominance']

trace1 = go.Scatter3d(
    x=x1,
    y=y1,
    z=z1,
```

```

    mode='markers',
    marker=dict(
        size=5,
        color='blue'
    ),
    name='The Daily Mail'
)

trace2 = go.Scatter3d(
    x=x2,
    y=y2,
    z=z2,
    mode='markers',
    marker=dict(
        size=5,
        color='red'
    ),
    name='The Guardian'
)

fig = go.Figure(data=[trace1, trace2])
fig.update_layout(
    title='Comparing VAD Scores Across The Daily Mail & The Guardian\'s Most Viewed Titles (On YouTube)',
    scene=dict(
        xaxis=dict(title='Valence'),
        yaxis=dict(title='Arousal'),
        zaxis=dict(title='Dominance')
    )
)
fig.show()

#File saved as html for your convenience.
fig.write_html("Comparing VAD Scores Across The Daily Mail & The Guardian\'s Most Viewed Titles (On YouTube).html")

#VAD values are visualised clearly in the 3D plot below. They seem to be very similar across all variables.
#They both also follow the very common positive linear relationship between Valence and Dominance.
#However, neither the DM nor TG use words low on Valence and Arousal.
#Perhaps an unarousing, sad story is not attractive to YouTube's fleeting audience base.
#Who may be attracted to a more upbeat tone.

```

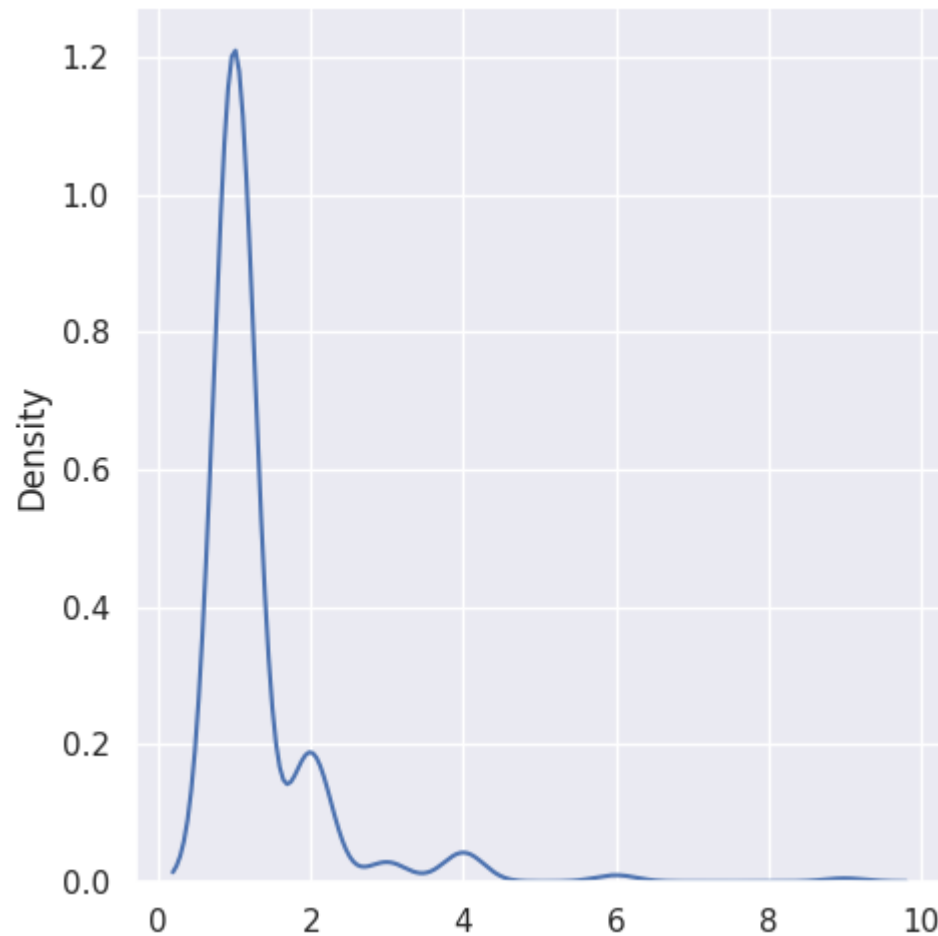
```
In [443... #ANALYSIS OF VOCABULARY RANGE

#THE GUARDIAN
#TG Plot counts
sns.displot(TG_word_counts, kind = 'kde')
plt.show()

#The Guardian uses a small range of vocabulary.
```



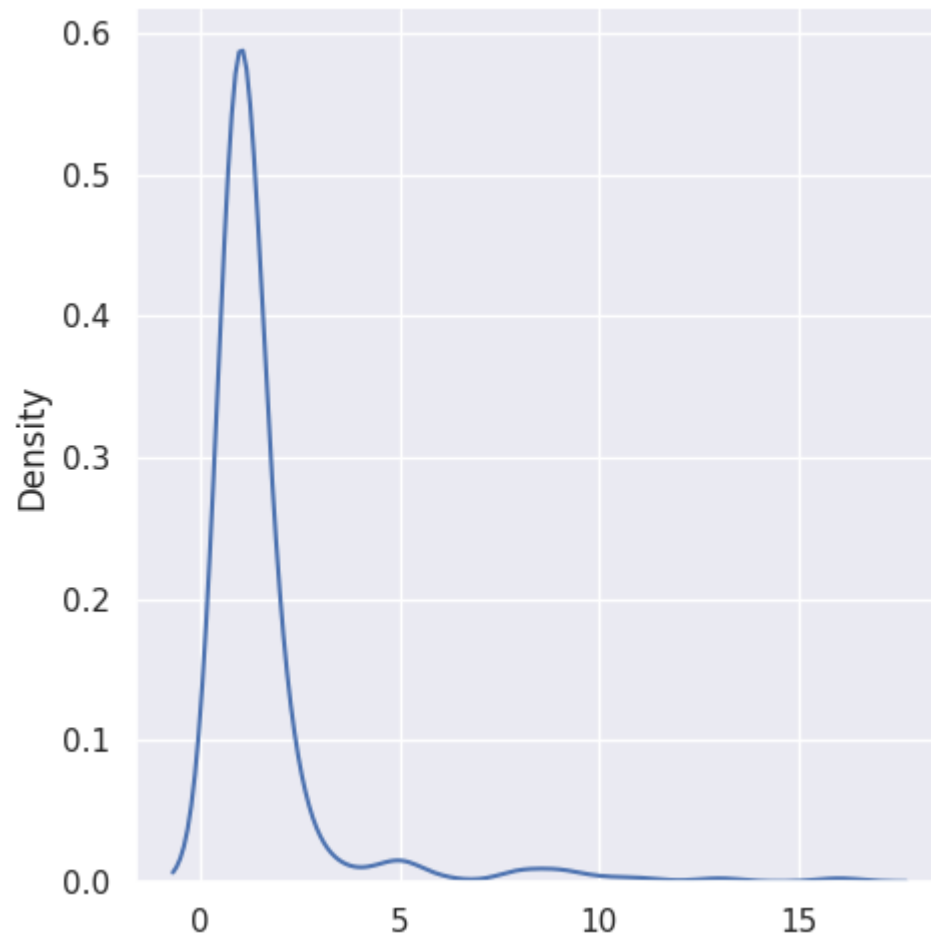
*#The distribution of vocabulary follows a shape alike to the usual Zipf's law (spike with right-skewed distribution)  
 #By frequenting the same words regularly, TG are able to communicate in a consistent, concise manner, through limit*



In [444... *#THE DAILY MAIL*

```
#DMPlot counts
sns.displot(DM_word_counts, kind = 'kde')
plt.show()
```

*#The DM features a vocabulary distribution alike Zipf's Law. However, it is wider in distribution than TG.  
 #DM use a more diverse vocabulary than TG.*



```
In [445... #PUNCTUATION ACROSS DM & TG

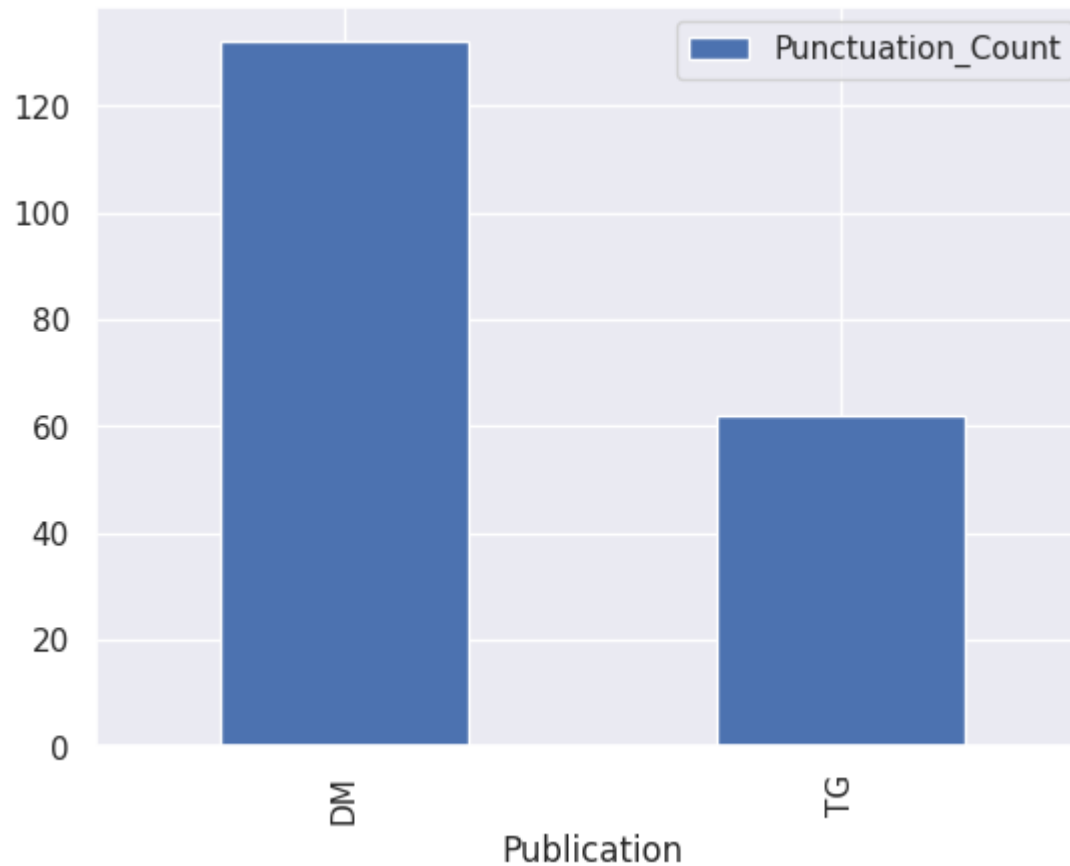
#DF
Punct_data = {'Publication': ['DM', 'TG'],
              'Punctuation_Count': [DM_num_punctuation, TG_num_punctuation]}
Punct_df = pd.DataFrame(Punct_data)

#.CSV file for your convenience
Punct_df.to_csv('Punctuation_Counts.csv', index=True)
```

```
In [446... #Plotting DF
Punct_df.plot(x='Publication', y='Punctuation_Count', kind='bar')

#This analysis demonstrates that DM uses more punctation in the their titles than TG.
#The difference is >100%! DM must believe that punctuation is an effective literary tool to grab audience attention
```

Out[446]: <Axes: xlabel='Publication'>



```
In [447... #UPPER CASE TEXT ACROSS DM & TG

#DF
UPPER_CASE_data = {'Publication': ['DM', 'TG'],
                   'UPPER_CASE_PROPORTION': [DM_PROP_UPPER, TG_PROP_UPPER]}
```

```
UPPER_CASE_df = pd.DataFrame(UPPER_CASE_data)

#.CSV file for your convenience
UPPER_CASE_df.to_csv('UPPER_CASE_PROPORTION.csv', index=True)
```

```
In [448]: #Plotting DF
UPPER_CASE_df.plot(x='Publication', y='UPPER_CASE_PROPORTION', kind='bar')

#DM uses more UPPER CASE TEXT in the their titles than TG.
#THIS INCREASES THE TITLES PRESENCE AND PERCEIVED URGENCY.
```

Out[448]: <Axes: xlabel='Publication'>



```
In [449... #.CSV file of Title dataset  
Combined_Titles_df = pd.DataFrame()  
Combined_Titles_df['TG_Titles'] = TG_title_list  
Combined_Titles_df['DM_Titles'] = DM_title_list  
Combined_Titles_df.to_csv('Combined_Titles.csv', index=True)  
  
#Source VAD values also available in folder.
```