CIS 545 Final Project -- Kevin Wang, Allison Zhang, Bhaskar Sen

Introduction:

Our group recognizes the mental health crisis currently sweeping through the globe, specifically within college-age students. Seeing this, we realized that indicators of poor mental health, specifically regarding things like depression, can be exhibited by users over the internet, such as tweets on Twitter. Furthermore, it is evident that the COVID-19 pandemic has exacerbated the mental health crisis to an even greater extent. Seeing this, we wanted to analyze a dataset containing COVID-19 related tweets to try and develop a model that could help predict for depressive sentiment.

However, we didn't want to just focus on English tweets. As Twitter has been growing in the past few years, its non-English speaking userbase has also grown, and been often less focused on in research studies. Thus, we wanted to focus on multi-lingual sentiment analysis, so that we could create a model that could predict depressive sentiment across a variety of languages, and then compare the results to detect any differences.

Our project will perform sentiment analysis on Tweets from the April of the beginning of the COVID-19 pandemic (https://www.kaggle.com/datasets/smid80/coronavirus-covid19-tweets-early-april). We hope to clean the data to a usable state, analyze sentiment across different languages and utilize different models to predict sentiment. By the end of the project, we will have a Zero Shot model that can take in text from multiple languages and output predictions matching the sentiment expressed with reasonably high accuracy.

Part 0: Imports and Set Up

0.1 Importing Basic Libraries and Setting up Kaggle

Here, we're importing many of the basic libraries that we'll have to use throughout this project, such as pandas, numpy, matplotlib, seaborn, etc.

Furthermore, we're also mounting the Google Drive folders and setting up Kaggle to download the datasets that we wish to use.

```
import json
import glob
import pandas as pd
import numpy as np
import datetime as dt
import re
```

```
import os
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import cm
from google.colab import drive

import numba
from numba import jit

!apt update
```

```
 0% [Working]              Hit:1 http://archive.ubuntu.com/ubuntu
bionic InRelease
 0% [Waiting for headers] [Waiting for headers] [Connected to cloud.r-
project.or
Hit:2 http://archive.ubuntu.com/ubuntu bionic-updates InRelease

Hit:3 http://security.ubuntu.com/ubuntu bionic-security InRelease

Hit:4 http://ppa.launchpad.net/c2d4u.team/c2d4u4.0+/ubuntu bionic
InRelease
 0% [Waiting for headers] [Connected to cloud.r-project.org
(65.9.86.118)] [Wait 0% [1 InRelease gpgv 242 kB] [Waiting for
headers] [Connected to cloud.r-projec
Hit:5 http://archive.ubuntu.com/ubuntu bionic-backports InRelease
 0% [1 InRelease gpgv 242 kB] [Waiting for headers] [Waiting for
headers] [Waiti
Hit:6 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran40/
InRelease

0% [1 InRelease gpgv 242 kB] [Waiting for headers] [Waiting for
headers]
Hit:7 http://ppa.launchpad.net/cran/libgit2/ubuntu bionic InRelease
 0% [1 InRelease gpgv 242 kB] [Waiting for headers] [Connecting to
ppa.launchpad
Hit:8 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu bionic InRelease
Hit:9 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu bionic
InRelease
Ign:10 https://developer.download.nvidia.com/compute/machine-
learning/repos/ubuntu1804/x86_64  InRelease
Hit:11
https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/
x86_64  InRelease
Hit:12 https://developer.download.nvidia.com/compute/machine-
learning/repos/ubuntu1804/x86_64  Release
Reading package lists... Done
Building dependency tree
Reading state information... Done
22 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

```
# Run this cell to mount your drive (you will be prompted to sign in)
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly
remount, call drive.mount("/content/drive", force_remount=True).
```

```
# Install kaggle through pip to allow downloading kaggle datasets
!pip install kaggle
```

```
Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: kaggle in
/usr/local/lib/python3.8/dist-packages (1.5.12)
Requirement already satisfied: tqdm in /usr/local/lib/python3.8/dist-
packages (from kaggle) (4.64.1)
Requirement already satisfied: requests in
/usr/local/lib/python3.8/dist-packages (from kaggle) (2.23.0)
Requirement already satisfied: six>=1.10 in
/usr/local/lib/python3.8/dist-packages (from kaggle) (1.15.0)
Requirement already satisfied: urllib3 in
/usr/local/lib/python3.8/dist-packages (from kaggle) (1.24.3)
Requirement already satisfied: python-dateutil in
/usr/local/lib/python3.8/dist-packages (from kaggle) (2.8.2)
Requirement already satisfied: certifi in
/usr/local/lib/python3.8/dist-packages (from kaggle) (2022.9.24)
Requirement already satisfied: python-slugify in
/usr/local/lib/python3.8/dist-packages (from kaggle) (7.0.0)
Requirement already satisfied: text-unidecode>=1.3 in
/usr/local/lib/python3.8/dist-packages (from python-slugify->kaggle)
(1.3)
Requirement already satisfied: chardet<4,>=3.0.2 in
/usr/local/lib/python3.8/dist-packages (from requests->kaggle) (3.0.4)
Requirement already satisfied: idna<3,>=2.5 in
/usr/local/lib/python3.8/dist-packages (from requests->kaggle) (2.10)
```

```
# Create the kaggle directory and read the uploaded kaggle.json file
# (NOTE: Do NOT run this cell more than once unless restarting kernel)
!mkdir ~/.kaggle
```

```
mkdir: cannot create directory ‘/root/.kaggle’: File exists
```

```
# Read the uploaded kaggle.json file
!cp /content/drive/MyDrive/kaggle.json ~/.kaggle/
```

```
# Download dataset
!!kaggle datasets download -d smid80/coronavirus-covid19-tweets-early-
april
```

```
['coronavirus-covid19-tweets-early-april.zip: Skipping, found more
recently modified local copy (use --force to force download)']
```

```
# Unzip folder in Colab content folder
!unzip /content/coronavirus-covid19-tweets-early-april.zip
```

```
Archive:  /content/coronavirus-covid19-tweets-early-april.zip
replace 2020-03-29 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-03-30 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-03-31 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-01 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-02 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-03 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-04 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-05 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-06 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-07 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-08 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-09 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-10 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-11 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
replace 2020-04-12 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
n
replace 2020-04-13 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: replace 2020-04-14 Coronavirus Tweets.CSV? [y]es, [n]o,
[A]ll, [N]one, [r]ename: n
replace 2020-04-15 Coronavirus Tweets.CSV? [y]es, [n]o, [A]ll, [N]one,
[r]ename: n
```

```
# Create the list of kaggle csv files using glob that were just
unzipped for later iteration
import os
import glob

filepath = "/content"
csv_list = glob.glob(filepath + "/*.CSV")
print(csv_list)
#df_march_29 = pd.read_csv('2020-03-29 Coronavirus Tweets.CSV')
```

```
['/content/2020-04-03 Coronavirus Tweets.CSV', '/content/2020-04-09
Coronavirus Tweets.CSV', '/content/2020-04-04 Coronavirus Tweets.CSV',
'/content/2020-04-11 Coronavirus Tweets.CSV', '/content/2020-03-30
Coronavirus Tweets.CSV', '/content/2020-04-14 Coronavirus Tweets.CSV',
'/content/2020-04-12 Coronavirus Tweets.CSV', '/content/2020-03-31
Coronavirus Tweets.CSV', '/content/2020-04-08 Coronavirus Tweets.CSV',
'/content/2020-04-13 Coronavirus Tweets.CSV', '/content/2020-04-07
Coronavirus Tweets.CSV', '/content/2020-04-02 Coronavirus Tweets.CSV',
'/content/2020-04-01 Coronavirus Tweets.CSV', '/content/2020-04-05
Coronavirus Tweets.CSV', '/content/2020-04-06 Coronavirus Tweets.CSV',
'/content/2020-04-10 Coronavirus Tweets.CSV', '/content/2020-04-15
Coronavirus Tweets.CSV', '/content/2020-03-29 Coronavirus Tweets.CSV']
```

Part I: Preprocessing, Exploratory Data Analysis, and Language Analysis Using Pandas

1.1: Data Loading

We want to create a Pandas Dataframe that contains the combined data from every single day within this kaggle dataset on COVID tweets during early April, ideally.

However, because the total dataframe that results from using all days results in a dataframe with over 8 million rows, for practicality's sake we chose to just analyze the first day's worth of data, namely data from March 29th, 2020. Else, running even basic language analysis functions ended up taking hours, which made coding and testing extremely impractical.

In the future, for further analysis, the entire dataset should be used. Beyond that, the same author on kaggle also released a follow-up dataset on COVID tweets in late April that, when combined with the results of this dataset, could be even more useful in revealing trends in sentiment.

The link to the follow-up dataset is here:
https://www.kaggle.com/datasets/smid80/coronavirus-covid19-tweets-late-april

```
# This is the overall dataframe, tweets_df, that contains all the
tweets
tweets_df = pd.DataFrame()

# Iterate through the list of csv's and add the temporary dataframe
per csv file to the overall dataframe
for csv in csv_list:
  temp_df = pd.read_csv(csv)
  tweets_df = pd.concat([tweets_df, temp_df])

  # This break line is used to only extract data from the 1st day
  # If more capability to run functions faster occurs, then remove
```

```
this!
  break

tweets_df
```

```
                  status_id              user_id            created_at
\
0       1245863586586439680            133184048  2020-04-03T00:00:00Z

1       1245863584417992704           2722502906  2020-04-03T00:00:00Z

2       1245863584799678464   775704843994353665  2020-04-03T00:00:00Z

3       1245863587307868160   860252856829587457  2020-04-03T00:00:00Z

4       1245863586892779523             88957440  2020-04-03T00:00:00Z

...                     ...                  ...                   ...

536152  1246225970887081984           1599189781  2020-04-03T23:59:59Z

536153  1246225970433941505            152835605  2020-04-03T23:59:59Z

536154  1246225970312462341           3820878372  2020-04-03T23:59:59Z

536155  1246225970379530245            374870492  2020-04-03T23:59:59Z

536156  1246225969586855936  1141543616453779458  2020-04-03T23:59:59Z


            screen_name
text  \
0             EcuadorTV  #QuédateEnCasa | Mira estas 𝐜𝐫𝐞𝐚𝐜𝐢𝐨𝐧𝐞𝐬
𝐨𝐫𝐢𝐠𝐢𝐧𝐚...
1          GradaNorteMX  Contra el #Coronavirus 😷👊🦠\n\n🏠
#QuédateEnCas...
2        AutoSupplyNews  @HondaMexico extiende suspensión de sus
planta...
3          IMSS_SanLuis  Con manos limpias, seguro estarás mejor.
#Prev...
4             Imagen_Mx  🚨😷 Baja California suma cuatro muertos por
#CO...
...                  ...
...
536152           SylJud  Last day of @GarrisonMillES Spirit Week. It
ma...
536153   thinkingautism  This is a real concern for our USian
community...
536154        dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
536155       cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
```

```
mundo,...
536156  YouIsWrongAgain  @nytimes Oh looky. now the #coronavirus is
rac...

                     source  reply_to_status_id  reply_to_user_id  \
0                 TweetDeck                 NaN               NaN
1                 TweetDeck                 NaN               NaN
2                 TweetDeck                 NaN      2.019508e+08
3                 TweetDeck                 NaN               NaN
4                 TweetDeck                 NaN               NaN
...                     ...                 ...               ...
536152   Twitter for iPhone                 NaN               NaN
536153            TweetDeck                 NaN               NaN
536154  Twitter for Android                 NaN               NaN
536155   Twitter for iPhone        1.246212e+18      7.026910e+17
536156   Twitter for iPhone        1.246201e+18      8.070950e+05

       reply_to_screen_name  is_quote  ...  retweet_count
country_code  \
0                       NaN     False  ...             15
NaN
1                       NaN     False  ...              0
NaN
2               HondaMexico     False  ...              0
NaN
3                       NaN     False  ...              0
NaN
4                       NaN     False  ...              1
NaN
...                     ...       ...  ...            ...            ..
.
536152                  NaN     False  ...              1
US
536153                  NaN     False  ...              8
NaN
536154                  NaN      True  ...              0
NaN
536155        PaisMenosPeor     False  ...              0
NaN
536156              nytimes     False  ...              0
NaN

        place_full_name place_type followers_count friends_count  \
0                   NaN        NaN          536720          1164
1                   NaN        NaN            1847           252
2                   NaN        NaN             538           780
3                   NaN        NaN            1015            41
4                   NaN        NaN          293891           269
...                 ...        ...             ...           ...
536152     Georgia, USA      admin             103           102
```

```
536153               NaN        NaN            40533           6687
536154               NaN        NaN               57             59
536155               NaN        NaN              349           1214
536156               NaN        NaN              105            212

        account_lang    account_created_at  verified lang
0                NaN  2010-04-15T06:31:39Z      True   es
1                NaN  2014-08-10T21:20:32Z     False   es
2                NaN  2016-09-13T14:37:01Z     False   es
3                NaN  2017-05-04T22:00:38Z     False   es
4                NaN  2009-11-10T16:01:41Z      True   es
...              ...                   ...       ...  ...
536152           NaN  2013-07-16T20:00:36Z     False   en
536153           NaN  2010-06-07T00:49:38Z     False   en
536154           NaN  2015-10-08T03:09:46Z     False   es
536155           NaN  2011-09-17T03:18:48Z     False   es
536156           NaN  2019-06-20T03:09:41Z     False   en

[536157 rows x 22 columns]
```

1.2: Language Analysis

As we want to do multi-lingual sentiment analysis, we have to investigate which languages are shared between all of the numerous libraries that are used for sentiment analysis. This includes important aspects like stemming, stopword usage, tokenizing, and which languages are supported by the zero-shot encoding models.

For analysis, we want to have a large enough sample of tweets in each language such that the analysis can be robust and more generally representative of a large number of users and their sentiment. Without this, just the presence of a few users could wildly swing the training model's classification of sentiment and emotion rather than having many users provide a solid baseline for training.

Thus, we want to find the total count of tweets per language, identify the languages with count above the median count of tweets per language, and keep those.

From there, we wish to only keep the languages that the stemming, stopword usage, tokenizing, and modeling also can use, so that when we analyze the multilingual sentiment we have the appropriate resources to process and generate sentiment properly.

1.2.1: Finding the Languages Used and Plotting Those With Counts Above Median

```
# We want to investigate which classes all of the columns are, along
with which seem to have a lot of nulls
# The results show that these columns have many nulls, so we should
keep an eye out for them:
# reply_to_status_id, reply_to_user_id, reply_to_screen_name,
```

```
country_code, place_full_name, place_type, and account_lang
tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 536157 entries, 0 to 536156
Data columns (total 22 columns):
 #   Column                Non-Null Count   Dtype  
---  ------                --------------   -----  
 0   status_id             536157 non-null  int64  
 1   user_id               536157 non-null  int64  
 2   created_at            536157 non-null  object 
 3   screen_name           536157 non-null  object 
 4   text                  536157 non-null  object 
 5   source                536155 non-null  object 
 6   reply_to_status_id    62903 non-null   float64
 7   reply_to_user_id      76988 non-null   float64
 8   reply_to_screen_name  76988 non-null   object 
 9   is_quote              536157 non-null  bool   
 10  is_retweet            536157 non-null  bool   
 11  favourites_count      536157 non-null  int64  
 12  retweet_count         536157 non-null  int64  
 13  country_code          24028 non-null   object 
 14  place_full_name       24105 non-null   object 
 15  place_type            24105 non-null   object 
 16  followers_count       536157 non-null  int64  
 17  friends_count         536157 non-null  int64  
 18  account_lang          0 non-null       float64
 19  account_created_at    536157 non-null  object 
 20  verified              536157 non-null  bool   
 21  lang                  536157 non-null  object 
dtypes: bool(3), float64(3), int64(6), object(10)
memory usage: 79.3+ MB
```

```
# We want to see which different languages are present within the
overall tweets_df dataframe
tweets_df.lang.unique()
```

```
array(['es', 'tl', 'en', 'in', 'pt', 'ar', 'ur', 'und', 'hi', 'si',
'fr',
       'uk', 'hu', 'am', 'tr', 'ja', 'it', 'cs', 'de', 'ca', 'fa',
'th',
       'is', 'zh', 'ru', 'el', 'ht', 'nl', 'ta', 'et', 'mr', 'pa',
'eu',
       'ko', 'sl', 'ro', 'da', 'pl', 'sr', 'vi', 'cy', 'sv', 'te',
'fi',
       'ne', 'lt', 'kn', 'ml', 'bn', 'ckb', 'iw', 'or', 'gu', 'no',
'ps',
       'km', 'sd', 'lv', 'dv', 'my', 'bg', 'ka', 'hy'], dtype=object)
```

```
# We add a column to tweets_df that has the total number of tweets
that language is written in
tweets_df['lang_count'] = tweets_df.groupby('lang')
['lang'].transform('count')
tweets_df
```

```
                  status_id              user_id            created_at
\
0       1245863586586439680            133184048  2020-04-03T00:00:00Z

1       1245863584417992704           2722502906  2020-04-03T00:00:00Z

2       1245863584799678464   775704843994353665  2020-04-03T00:00:00Z

3       1245863587307868160   860252856829587457  2020-04-03T00:00:00Z

4       1245863586892779523             88957440  2020-04-03T00:00:00Z

...                     ...                  ...                   ...

536152  1246225970887081984           1599189781  2020-04-03T23:59:59Z

536153  1246225970433941505            152835605  2020-04-03T23:59:59Z

536154  1246225970312462341           3820878372  2020-04-03T23:59:59Z

536155  1246225970379530245            374870492  2020-04-03T23:59:59Z

536156  1246225969586855936  1141543616453779458  2020-04-03T23:59:59Z


            screen_name
text  \
0             EcuadorTV  #QuédateEnCasa | Mira estas creaciones
origina...
1          GradaNorteMX  Contra el #Coronavirus 😒😷◯▯\n\n🏚
#QuédateEnCas...
2        AutoSupplyNews  @HondaMexico extiende suspensión de sus
planta...
3          IMSS_SanLuis  Con manos limpias, seguro estarás mejor.
#Prev...
4             Imagen_Mx  🚨😳 Baja California suma cuatro muertos por
#CO...
...                  ...
...
536152           SylJud  Last day of @GarrisonMillES Spirit Week. It
ma...
536153   thinkingautism  This is a real concern for our USian
community...
536154        dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
```

```
536155       cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
mundo,...
536156  YouIsWrongAgain  @nytimes Oh looky. now the #coronavirus is
rac...

                     source  reply_to_status_id  reply_to_user_id  \
0                 TweetDeck                 NaN               NaN
1                 TweetDeck                 NaN               NaN
2                 TweetDeck                 NaN      2.019508e+08
3                 TweetDeck                 NaN               NaN
4                 TweetDeck                 NaN               NaN
...                     ...                 ...               ...
536152   Twitter for iPhone                 NaN               NaN
536153            TweetDeck                 NaN               NaN
536154  Twitter for Android                 NaN               NaN
536155   Twitter for iPhone        1.246212e+18      7.026910e+17
536156   Twitter for iPhone        1.246201e+18      8.070950e+05

       reply_to_screen_name  is_quote  ...  country_code
place_full_name  \
0                       NaN     False  ...           NaN
NaN
1                       NaN     False  ...           NaN
NaN
2               HondaMexico     False  ...           NaN
NaN
3                       NaN     False  ...           NaN
NaN
4                       NaN     False  ...           NaN
NaN
...                     ...       ...  ...           ...
...
536152                  NaN     False  ...            US     Georgia,
USA
536153                  NaN     False  ...           NaN
NaN
536154                  NaN      True  ...           NaN
NaN
536155        PaisMenosPeor     False  ...           NaN
NaN
536156              nytimes     False  ...           NaN
NaN

        place_type followers_count friends_count account_lang  \
0              NaN          536720          1164          NaN
1              NaN            1847           252          NaN
2              NaN             538           780          NaN
3              NaN            1015            41          NaN
4              NaN          293891           269          NaN
...            ...             ...           ...          ...
```

```
536152       admin             103            102           NaN
536153         NaN           40533           6687           NaN
536154         NaN              57             59           NaN
536155         NaN             349           1214           NaN
536156         NaN             105            212           NaN

          account_created_at  verified  lang lang_count
0       2010-04-15T06:31:39Z      True    es      82558
1       2014-08-10T21:20:32Z     False    es      82558
2       2016-09-13T14:37:01Z     False    es      82558
3       2017-05-04T22:00:38Z     False    es      82558
4       2009-11-10T16:01:41Z      True    es      82558
...                      ...       ...   ...        ...
536152  2013-07-16T20:00:36Z     False    en     302604
536153  2010-06-07T00:49:38Z     False    en     302604
536154  2015-10-08T03:09:46Z     False    es      82558
536155  2011-09-17T03:18:48Z     False    es      82558
536156  2019-06-20T03:09:41Z     False    en     302604

[536157 rows x 23 columns]
```

```
# We take just the language column, turn it into its own dataframe,
and sort by the count
language_count = tweets_df.groupby('lang')
['lang'].count().reset_index(name='lang_count')
language_count = language_count.sort_values(by='lang_count',
ascending=True)
language_count
```

```
   lang  lang_count
29   ka           1
37   my           3
30   km           4
23   hy           7
5   ckb          11
..  ...         ...
26   it       13871
59  und       26589
18   fr       30123
13   es       82558
12   en      302604

[63 rows x 2 columns]
```

```
# We add the logarithmic language count column to the table to better
visualize outliers when we plot it.
language_count['log_lang_count'] =
np.log(language_count['lang_count'])
language_count
```

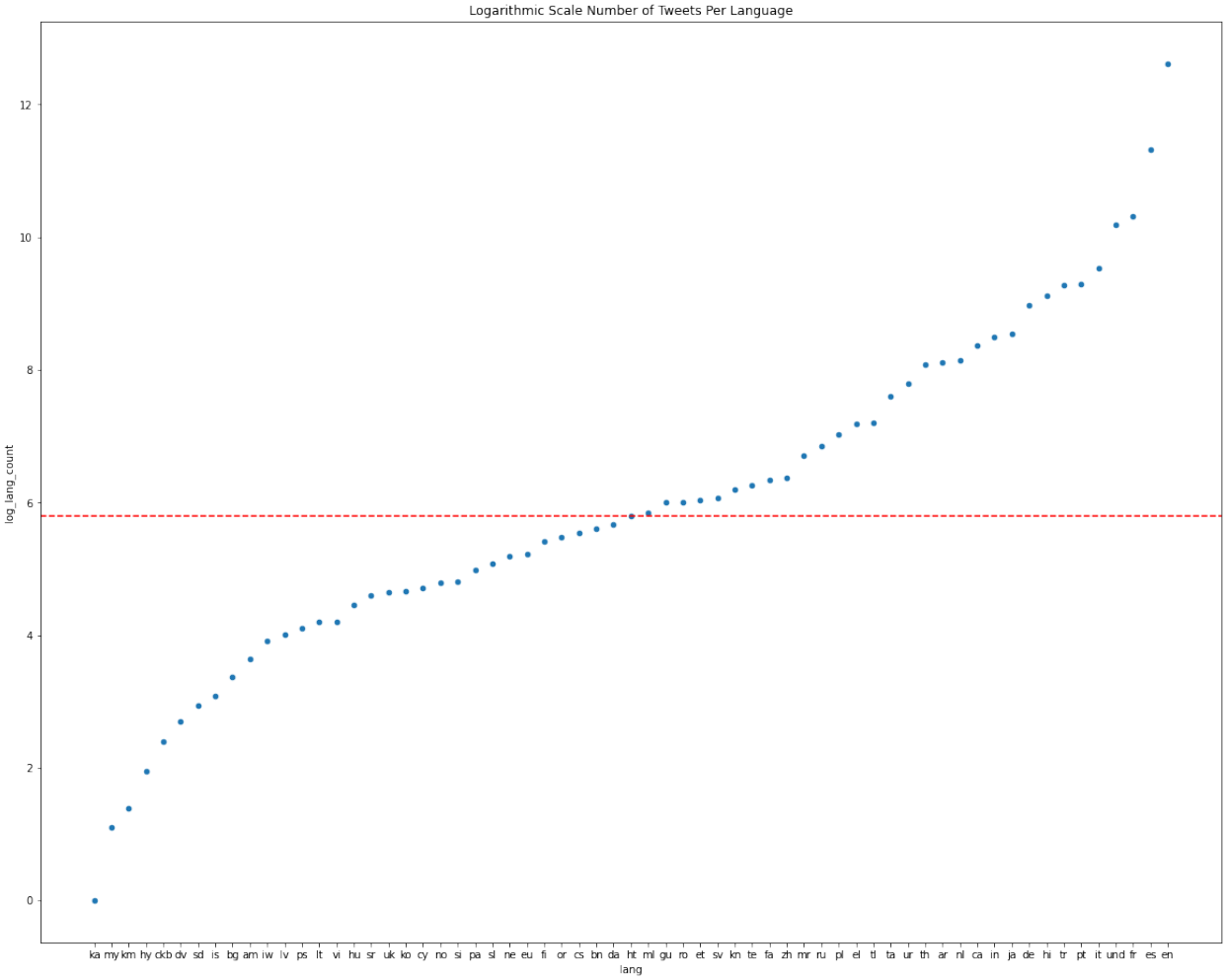
```
   lang  lang_count  log_lang_count
29   ka           1        0.000000
37   my           3        1.098612
30   km           4        1.386294
23   hy           7        1.945910
5   ckb          11        2.397895
..  ...         ...             ...
26   it       13871        9.537556
59  und       26589       10.188253
18   fr       30123       10.313044
13   es       82558       11.321256
12   en      302604       12.620180

[63 rows x 3 columns]
```

```
# We take the median language count for the logarithmic language
column, so we can graph it there
# Mean language count is NOT used because of the skewed distribution
making it inaccurate
median_language_count_logarithmic =
language_count['log_lang_count'].median()
print(median_language_count_logarithmic)
```

```
5.796057750765372
```

```
# We plot a scatterplot using the languages and log_lang_count
language_count.plot(kind='scatter', x='lang', y='log_lang_count',
figsize=(20, 16))
# We also plot a line at the median language count to show which are
above median
plt.axhline(y=median_language_count_logarithmic, color='r',
linestyle='--')
plt.title("Logarithmic Scale Number of Tweets Per Language")
```

```
Text(0.5, 1.0, 'Logarithmic Scale Number of Tweets Per Language')
```



This scatterplot shows the logarithmic distribution of the languages and the general (logarithmically transformed) count of tweets per language. The horizontal line indicates the median halfway point, and we will be considering only languages above the halfway point as so to ensure adequate sampling size for our analysis.

```
# This finds the median of the language counts without the logarithmic transformation applied
median_lang_count_normal =  language_count['lang_count'].median()
print(median_lang_count_normal)

329.0

#language_count[language_count['lang'] == 'ht']['lang_count'] This is for the overall tweets_df
language_count[language_count['lang'] == 'sv']['lang_count']

52    435
Name: lang_count, dtype: int64
```

```
#sufficient_language_df = language_count[language_count['lang_count']
> 5408] 5408 is for the overall one

# This puts all of the languages that have a language count above the
median into a separate dataframe to keep
sufficient_language_df = language_count[language_count['lang_count'] >
median_lang_count_normal]
sufficient_language_df
```

	lang	lang_count	log_lang_count
35	ml	347	5.849325
19	gu	405	6.003887
46	ro	408	6.011267
14	et	416	6.030685
52	sv	435	6.075346
31	kn	494	6.202536
54	te	521	6.255750
16	fa	570	6.345636
62	zh	581	6.364751
36	mr	813	6.700731
47	ru	946	6.852243
43	pl	1122	7.022868
11	el	1321	7.186144
56	tl	1340	7.200425
53	ta	2009	7.605392
60	ur	2418	7.790696
55	th	3220	8.077137
1	ar	3353	8.117611
39	nl	3432	8.140898
4	ca	4299	8.366138
24	in	4902	8.497399
28	ja	5140	8.544808
9	de	7846	8.967759
20	hi	9106	9.116689
57	tr	10650	9.273315
45	pt	10834	9.290445
26	it	13871	9.537556
59	und	26589	10.188253
18	fr	30123	10.313044
13	es	82558	11.321256
12	en	302604	12.620180

```
# This transforms those languages with above-median counts into a list
for set intersection
sufficient_language_list = sufficient_language_df['lang']
sufficient_language_list
```

```
35     ml
19     gu
46     ro
```

```
14     et
52     sv
31     kn
54     te
16     fa
62     zh
36     mr
47     ru
43     pl
11     el
56     tl
53     ta
60     ur
55     th
1      ar
39     nl
4      ca
24     in
28     ja
9      de
20     hi
57     tr
45     pt
26     it
59    und
18     fr
13     es
12     en
Name: lang, dtype: object
```

The languages that included in the tweets_df as indicated above; the ISO Language Codes are indicated in the parenthesis to the right of the language:

- Bangla (bn)
- Swedish (sv)
- Estonian (et)
- Malayalam (ml)
- Romanian (ro)
- Gujarati (gu)
- Kannada (kn)
- Chinese (zh)
- Telugu (te)
- Persian (fa)
- Marathi (mr)
- Russian (ru)
- Polish (pl)
- Tagalog (tl)

- Greek (el)
- Urdu (ur)
- Tamil (ta)
- Thai (th)
- Dutch (nl)
- Arabic (ar)
- Catalan (ca)
- Japanese (ja)
- Indonesian (in)
- German (de)
- Hindi (hi)
- Turkish (tr)
- Portuguese (pt)
- Italian (it)
- Undetermined (und)
- French (fr)
- Spanish (es)
- English (en)

These languages will be compared to the languages used by the various stemming/tokenizing/etc libraries to see which overlap and thus will be used for sentiment analysis.

```
# NLTK Tokenizer supported languages
# List: 'czech', 'danish', 'dutch', 'english', 'estonian', 'finnish',
'french', 'german', 'greek',
#       'italian', 'norwegian', 'polish', 'portuguese', 'russian',
'slovene', 'spanish', 'swedish', 'turkish'

tokenizer_supported_languages = ['cs', 'da', 'nl', 'en', 'et', 'fi',
'fr', 'de',
                                 'el', 'it', 'no', 'pl', 'pt', 'ru',
'sl', 'es', 'sv', 'tr']

# NLTK Stopwords languages
# List: 'arabic', 'azerbaijani', 'danish', 'dutch', 'english',
'finnish', 'french', 'german', 'greek', 'hungarian', 'indonesian',
#       'italian', 'kazakh', 'nepali', 'norwegian', 'portuguese',
'romanian', 'russian', 'slovene', 'spanish', 'swedish', 'tajik',
'turkish'

stopwords_supported_languages = ['ar', 'az', 'da', 'nl', 'en', 'fi',
'fr', 'de',
                                 'el', 'hu', 'in', 'it', 'kk', 'ne',
'no', 'pt', 'ro',
                                 'ru', 'sl', 'es', 'sv', 'tg', 'tr']
```

```
# SnowballStemmer supported languages
# List: 'arabic', 'danish', 'dutch', 'english', 'finnish', 'french',
'german',
#       'hungarian', 'italian', 'norwegian', 'porter', 'portuguese',
'romanian', 'russian', 'spanish', 'swedish'

stemmer_supported_languages = ['ar', 'da', 'nl', 'en', 'fi', 'fr',
'de',
                               'hu', 'it', 'no', 'pt', 'ro', 'ru',
'es', 'sv']

# XLM-Roberta-Large-XNLI Zero-Shot Modeling supported languages
# List: 'english', 'french', 'spanish', 'german', 'greek',
'bulgarian', 'russian',
# 'turkish', 'arabic', 'vietnamese', 'thai', 'chinese', 'hindi',
'swahili', 'urdu'

model_supported_languages = ['en', 'fr', 'es', 'de', 'el', 'bg', 'ru',
'tr',
                             'ar', 'vi', 'th', 'ch', 'hi', 'sw', 'ur']

# This uses set intersection to find which languages are shared by all
sets, and therefore by each toolkit/library.
fully_supported_languages =
set(tokenizer_supported_languages).intersection(set(stopwords_supporte
d_languages))
fully_supported_languages =
fully_supported_languages.intersection(set(stemmer_supported_languages
))
fully_supported_languages =
fully_supported_languages.intersection(set(model_supported_languages))
fully_supported_languages
```

```
{'de', 'en', 'es', 'fr', 'ru'}
```

Now we find the languages that have support in Stemming, Stopwords, and Tokenization, along with having over the median (5408 for all tweets, ~313 for one day) total tweet count through set intersections of all of them.

```
# This uses set intersection to also check that the languages
supported have sufficient data within our sample to analyze.
fully_supported_languages =
fully_supported_languages.intersection(set(sufficient_language_list))
fully_supported_languages
```

```
{'de', 'en', 'es', 'fr', 'ru'}
```

However, finding specific libraries that find compound, lexicon-based sentiment is difficult, as multi-lingual sentiment is still a rapidly developing and emerging field. The mass majority of

toolkits developed so far are for the English language, but finding toolkits for sentiment analysis of other languages proved to be much more difficult to do.

Through our analysis, we found compound, lexicon-based sentiment analysis toolkits of 2 other languages primarily, Spanish and French. There weren't any results for Russian sentiment analysis, and the primary German sentiment analysis toolkit we could find was this one: https://github.com/oliverguhr/german-sentiment-lib

However, the results of this one returned sentiment as ['negative', 'positive'], rather than a number, and thus complicated our analysis, so we decided to not use German inside of our analysis.

Thus, with our dataframes to be created, we chose to make them for the primary 3 languages supported and with sufficient data: English, Spanish, and French.

```
# These are the languages that lexicon-based, used in academic study
toolkits could be found for.
sentiment_supported_languages = ['en', 'es', 'fr']

# Thus, the only remaining languages to make datasets out of and
analyze are English, Spanish, and French.
fully_supported_languages =
fully_supported_languages.intersection(set(sentiment_supported_languag
es))
fully_supported_languages
```

```
{'en', 'es', 'fr'}
```

```
# This gets only the English tweets from tweets_df into a dataframe,
english_tweets_df
english_tweets_df = tweets_df[tweets_df['lang'] == 'en']
english_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 302604 entries, 8 to 536156
Data columns (total 23 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   status_id             302604 non-null  int64
 1   user_id               302604 non-null  int64
 2   created_at            302604 non-null  object
 3   screen_name           302604 non-null  object
 4   text                  302604 non-null  object
 5   source                302602 non-null  object
 6   reply_to_status_id    40284 non-null   float64
 7   reply_to_user_id      50142 non-null   float64
 8   reply_to_screen_name  50142 non-null   object
 9   is_quote              302604 non-null  bool
 10  is_retweet            302604 non-null  bool
 11  favourites_count      302604 non-null  int64
 12  retweet_count         302604 non-null  int64
```

```
 13  country_code           13999 non-null   object
 14  place_full_name        14055 non-null   object
 15  place_type             14055 non-null   object
 16  followers_count        302604 non-null  int64
 17  friends_count          302604 non-null  int64
 18  account_lang           0 non-null       float64
 19  account_created_at     302604 non-null  object
 20  verified               302604 non-null  bool
 21  lang                   302604 non-null  object
 22  lang_count             302604 non-null  int64
dtypes: bool(3), float64(3), int64(7), object(10)
memory usage: 49.3+ MB
```

```
# This gets only the Spanish tweets from tweets_df into a dataframe,
spanish_tweets_df
spanish_tweets_df = tweets_df[tweets_df['lang'] == 'es']
spanish_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 82558 entries, 0 to 536155
Data columns (total 23 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   status_id              82558 non-null  int64
 1   user_id                82558 non-null  int64
 2   created_at             82558 non-null  object
 3   screen_name            82558 non-null  object
 4   text                   82558 non-null  object
 5   source                 82558 non-null  object
 6   reply_to_status_id     7809 non-null   float64
 7   reply_to_user_id       9397 non-null   float64
 8   reply_to_screen_name   9397 non-null   object
 9   is_quote               82558 non-null  bool
 10  is_retweet             82558 non-null  bool
 11  favourites_count       82558 non-null  int64
 12  retweet_count          82558 non-null  int64
 13  country_code           3565 non-null   object
 14  place_full_name        3566 non-null   object
 15  place_type             3566 non-null   object
 16  followers_count        82558 non-null  int64
 17  friends_count          82558 non-null  int64
 18  account_lang           0 non-null      float64
 19  account_created_at     82558 non-null  object
 20  verified               82558 non-null  bool
 21  lang                   82558 non-null  object
 22  lang_count             82558 non-null  int64
dtypes: bool(3), float64(3), int64(7), object(10)
memory usage: 13.5+ MB
```

```
# This gets only the French tweets from tweets_df into a dataframe,
french_tweets_df
french_tweets_df = tweets_df[tweets_df['lang'] == 'fr']
french_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 30123 entries, 141 to 536146
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   status_id             30123 non-null  int64
 1   user_id               30123 non-null  int64
 2   created_at            30123 non-null  object
 3   screen_name           30123 non-null  object
 4   text                  30123 non-null  object
 5   source                30123 non-null  object
 6   reply_to_status_id    3395 non-null   float64
 7   reply_to_user_id      3932 non-null   float64
 8   reply_to_screen_name  3932 non-null   object
 9   is_quote              30123 non-null  bool
 10  is_retweet            30123 non-null  bool
 11  favourites_count      30123 non-null  int64
 12  retweet_count         30123 non-null  int64
 13  country_code          945 non-null    object
 14  place_full_name       946 non-null    object
 15  place_type            946 non-null    object
 16  followers_count       30123 non-null  int64
 17  friends_count         30123 non-null  int64
 18  account_lang          0 non-null      float64
 19  account_created_at    30123 non-null  object
 20  verified              30123 non-null  bool
 21  lang                  30123 non-null  object
 22  lang_count            30123 non-null  int64
dtypes: bool(3), float64(3), int64(7), object(10)
memory usage: 4.9+ MB
```

```
german_tweets_df = tweets_df[tweets_df['lang'] == 'de']
german_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 7846 entries, 690 to 536031
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   status_id             7846 non-null   int64
 1   user_id               7846 non-null   int64
 2   created_at            7846 non-null   object
 3   screen_name           7846 non-null   object
 4   text                  7846 non-null   object
 5   source                7846 non-null   object
```

```
 6   reply_to_status_id    925 non-null    float64
 7   reply_to_user_id      1026 non-null   float64
 8   reply_to_screen_name  1026 non-null   object
 9   is_quote              7846 non-null   bool
 10  is_retweet            7846 non-null   bool
 11  favourites_count      7846 non-null   int64
 12  retweet_count         7846 non-null   int64
 13  country_code          207 non-null    object
 14  place_full_name       207 non-null    object
 15  place_type            207 non-null    object
 16  followers_count       7846 non-null   int64
 17  friends_count         7846 non-null   int64
 18  account_lang          0 non-null      float64
 19  account_created_at    7846 non-null   object
 20  verified              7846 non-null   bool
 21  lang                  7846 non-null   object
 22  lang_count            7846 non-null   int64
dtypes: bool(3), float64(3), int64(7), object(10)
memory usage: 1.3+ MB
```

1.3: English Tweets Analysis

1.3.1: English Tweets Pre-processing and EDA

Now, we want to investigate the English tweets and remove redundant/unnecessary columns, while finding any preliminary patterns and/or correlations within the dataset that could be of interest.

```
# Looks at the first 20 rows of the dataframe to get a better idea of
what it looks like.
english_tweets_df.head(20)
```

```
              status_id              user_id            created_at  \
8   1245863586892636161             36969470  2020-04-03T00:00:00Z
10  1245863584447463425             36327407  2020-04-03T00:00:00Z
13  1245863586989248512            243811680  2020-04-03T00:00:00Z
14  1245863585227714560             44728980  2020-04-03T00:00:00Z
16  1245863584355307522            171548670  2020-04-03T00:00:00Z
26  1245863586838114304   747891181338525696  2020-04-03T00:00:00Z
29  1245863586322210817             18349059  2020-04-03T00:00:00Z
32  1245863587517730822  1079061579973439488  2020-04-03T00:00:00Z
33  1245863587404382209   748641810327605248  2020-04-03T00:00:00Z
35  1245863587144298498             27922157  2020-04-03T00:00:00Z
36  1245863587140104193             47798025  2020-04-03T00:00:00Z
37  1245863586892853249            242772524  2020-04-03T00:00:00Z
39  1245863586368499712           1275136292  2020-04-03T00:00:00Z
42  1245863585814896640             64643056  2020-04-03T00:00:00Z
43  1245863585781149697            282176072  2020-04-03T00:00:00Z
45  1245863585722425345             34042766  2020-04-03T00:00:00Z
```

```
48  1245863585630191616           1888044565  2020-04-03T00:00:00Z
51  1245863585345085443            571878474  2020-04-03T00:00:00Z
53  1245863585252691973   746143379197591552  2020-04-03T00:00:00Z
54  1245863585177317381  1065029298799529984  2020-04-03T00:00:00Z

        screen_name                                                text
\
8      BostonReview  "The constantly shifting nature of a public he...

10         htTweets  Iran parliament speaker tests positive for #CO...

13        CBS4Local  “Domestic violence did not start during #COVID...

14        ANCALERTS  The shows must go on!: Lloyd Webber musicals t...

16      RadioNLNews  BC's Medical Health Officer says the province ...

26  cosmicfirepeace  #trump claims he's saved 1,980,000 #Americans ...

29      RedMaryland  Red Maryland Radio is LIVE with @BrianGriffith...

32      RepSusieLee  Thank you to our child care providers during #...

33    ingresscanada  #Canadian Educational Institutions might suffe...

35  AmerMedicalAssn  In this #COVID19 update, #OurAMA experts discu...

36       EBmedicine  New evidence and advice on SARS-CoV-2 testing ...

37    RepFrankLucas  Our nation's small businesses are facing an un...

39      fams2gether  Have medical supplies you can donate during #C...

42           RT_com  #Cancun police spread #COVID19 awareness with ...

43       MFSAVoices  Overseas Filipinos and supporters condemn Dute...

45         McKinsey  The #COVID19 pandemic is a threat not only to ...

48          CCPA_BC  Hear @1alexhemingway in conversation with @Jef...

51       DDNational  PLEASE RETWEET -\n\nHere is a list of DOs you ...

53   CHSCAlamedaCty  It's important to stay safe  so that we can re...

54      SupFletcher  Hundreds of languages are spoken in D4, and we...


                      source  reply_to_status_id  reply_to_user_id  \
8                     Buffer                 NaN               NaN
10                 TweetDeck                 NaN               NaN
```

```
13                   TweetDeck                   NaN                  NaN
14                   TweetDeck                   NaN                  NaN
16                   TweetDeck                   NaN                  NaN
26             Twitter Web App                   NaN                  NaN
29                   TweetDeck                   NaN                  NaN
32                   TweetDeck                   NaN                  NaN
33                  Zoho Social                  NaN                  NaN
35                     Sprinklr                  NaN                  NaN
36  SEMrush Social Media Tool                    NaN                  NaN
37                   TweetDeck                   NaN                  NaN
39                   TweetDeck                   NaN                  NaN
42        Twitter Media Studio                   NaN                  NaN
43                   TweetDeck                   NaN                  NaN
45                    Sprinklr                   NaN                  NaN
48                   TweetDeck                   NaN                  NaN
51        Twitter Media Studio                   NaN                  NaN
53                   TweetDeck                   NaN                  NaN
54                   TweetDeck                   NaN                  NaN

   reply_to_screen_name  is_quote  ...  country_code  place_full_name
\
8                   NaN     False  ...           NaN               NaN

10                  NaN     False  ...           NaN               NaN

13                  NaN     False  ...           NaN               NaN

14                  NaN     False  ...           NaN               NaN

16                  NaN     False  ...           NaN               NaN

26                  NaN     False  ...           NaN               NaN

29                  NaN     False  ...           NaN               NaN

32                  NaN     False  ...           NaN               NaN

33                  NaN     False  ...           NaN               NaN

35                  NaN     False  ...           NaN               NaN

36                  NaN     False  ...           NaN               NaN

37                  NaN     False  ...           NaN               NaN

39                  NaN     False  ...           NaN               NaN

42                  NaN     False  ...           NaN               NaN

43                  NaN     False  ...           NaN               NaN
```

```
45                  NaN     False  ...           NaN               NaN

48                  NaN     False  ...           NaN               NaN

51                  NaN     False  ...           NaN               NaN

53                  NaN     False  ...           NaN               NaN

54                  NaN     False  ...           NaN               NaN


    place_type followers_count friends_count account_lang  \
8          NaN           42537          2761          NaN
10         NaN         7249747           125          NaN
13         NaN           13438           510          NaN
14         NaN         4889749           776          NaN
16         NaN            6934          2137          NaN
26         NaN            3859          4999          NaN
29         NaN            5045           447          NaN
32         NaN           16692           229          NaN
33         NaN              14            28          NaN
35         NaN          716974          6890          NaN
36         NaN            7056          7215          NaN
37         NaN           20761           319          NaN
39         NaN           36130          1272          NaN
42         NaN         3074547           640          NaN
43         NaN            5436          3550          NaN
45         NaN          375519          1174          NaN
48         NaN            5169          1711          NaN
51         NaN          427105           354          NaN
53         NaN              82           156          NaN
54         NaN            2504           639          NaN

      account_created_at  verified  lang lang_count
8   2009-05-01T15:42:57Z     False    en     302604
10  2009-04-29T10:11:34Z      True    en     302604
13  2011-01-27T21:33:02Z      True    en     302604
14  2009-06-04T21:26:24Z      True    en     302604
16  2010-07-27T16:17:02Z     False    en     302604
26  2016-06-28T20:35:27Z     False    en     302604
29  2008-12-24T02:58:16Z     False    en     302604
32  2018-12-29T17:08:23Z      True    en     302604
33  2016-06-30T22:18:11Z     False    en     302604
35  2009-03-31T17:50:31Z      True    en     302604
36  2009-06-17T00:58:22Z     False    en     302604
37  2011-01-25T15:24:07Z      True    en     302604
39  2013-03-17T14:38:49Z      True    en     302604
42  2009-08-11T06:12:45Z      True    en     302604
43  2011-04-14T18:03:16Z     False    en     302604
45  2009-04-21T21:09:11Z      True    en     302604
```

```
48  2013-09-20T22:22:32Z     False    en     302604
51  2012-05-05T14:36:20Z      True    en     302604
53  2016-06-24T00:50:19Z     False    en     302604
54  2018-11-20T23:49:06Z     False    en     302604

[20 rows x 23 columns]
```

```
# Looks at the info summary of english_tweets_df to discover the
columns with high amounts of nulls
# Just like in tweets_df, those specific columns indicated as
problematic have many nulls!
english_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 302604 entries, 8 to 536156
Data columns (total 23 columns):
 #   Column                Non-Null Count   Dtype
---  ------                --------------   -----
 0   status_id             302604 non-null  int64
 1   user_id               302604 non-null  int64
 2   created_at            302604 non-null  object
 3   screen_name           302604 non-null  object
 4   text                  302604 non-null  object
 5   source                302602 non-null  object
 6   reply_to_status_id    40284 non-null   float64
 7   reply_to_user_id      50142 non-null   float64
 8   reply_to_screen_name  50142 non-null   object
 9   is_quote              302604 non-null  bool
 10  is_retweet            302604 non-null  bool
 11  favourites_count      302604 non-null  int64
 12  retweet_count         302604 non-null  int64
 13  country_code          13999 non-null   object
 14  place_full_name       14055 non-null   object
 15  place_type            14055 non-null   object
 16  followers_count       302604 non-null  int64
 17  friends_count         302604 non-null  int64
 18  account_lang          0 non-null       float64
 19  account_created_at    302604 non-null  object
 20  verified              302604 non-null  bool
 21  lang                  302604 non-null  object
 22  lang_count            302604 non-null  int64
dtypes: bool(3), float64(3), int64(7), object(10)
memory usage: 49.3+ MB
```

1.3.1.1: Drop Nulls in English Tweets

Looking at these, we drop the columns such as reply_to_user_id or place_full_name that have over 75% null values.

These have an extremely large number of null values, for one, and cannot be easily imputated!

For example, if the country code or place is null, there's no easy imputation to be had.

Likewise, whether or not it's a reply is rather meaningless in our analysis, so we can drop it rather than doing null imputation!

```
# Drops columns with null value count above 75% of the total number of
rows.
english_tweets_df.dropna(thresh=int(0.75 *
english_tweets_df.shape[0]), axis=1, inplace=True)
english_tweets_df
```

```
/usr/local/lib/python3.8/dist-packages/pandas/util/_decorators.py:311:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return func(*args, **kwargs)
```

	status_id	user_id	created_at	screen_name	text
8	1245863586892636161	36969470	2020-04-03T00:00:00Z	BostonReview	"The constantly shifting nature of a public he...
10	1245863584447463425	36327407	2020-04-03T00:00:00Z	htTweets	Iran parliament speaker tests positive for #CO...
13	1245863586989248512	243811680	2020-04-03T00:00:00Z		
14	1245863585227714560	44728980	2020-04-03T00:00:00Z		
16	1245863584355307522	171548670	2020-04-03T00:00:00Z		
...	...	...	...		
536149	1246225964729679872	1246079496635043845	2020-04-03T23:59:58Z		
536151	1246225965518372865	128713921	2020-04-03T23:59:58Z		
536152	1246225970887081984	1599189781	2020-04-03T23:59:59Z		
536153	1246225970433941505	152835605	2020-04-03T23:59:59Z		
536156	1246225969586855936	1141543616453779458	2020-04-03T23:59:59Z		

```
13             CBS4Local  “Domestic violence did not start during
#COVID...
14             ANCALERTS  The shows must go on!: Lloyd Webber musicals
t...
16           RadioNLNews  BC's Medical Health Officer says the province
...
...                   ...
...
536149       RoseTacazon  "These days of pain and sadness are bringing
m...
536151        XopherKyle  Did #POTUS just lie that many states operate
D...
536152            SylJud  Last day of @GarrisonMillES Spirit Week. It
ma...
536153   thinkingautism  This is a real concern for our USian
community...
536156  YouIsWrongAgain  @nytimes Oh looky. now the #coronavirus is
rac...

                     source  is_quote  is_retweet  favourites_count  \
8                    Buffer     False       False             16531
10                TweetDeck     False       False              2975
13                TweetDeck     False       False               495
14                TweetDeck     False       False              5450
16                TweetDeck     False       False               500
...                     ...       ...         ...                ...
536149  Twitter for Android     False       False                 4
536151   Twitter for iPhone     False       False             10978
536152   Twitter for iPhone     False       False               699
536153            TweetDeck     False       False              9373
536156   Twitter for iPhone     False       False              4633

        retweet_count  followers_count  friends_count
account_created_at  \
8                   0            42537           2761  2009-05-
01T15:42:57Z
10                  4          7249747            125  2009-04-
29T10:11:34Z
13                  0            13438            510  2011-01-
27T21:33:02Z
14                  6          4889749            776  2009-06-
04T21:26:24Z
16                  0             6934           2137  2010-07-
27T16:17:02Z
...               ...              ...            ...
...
536149              0                0             29  2020-04-
03T14:18:07Z
536151              1             5531           5255  2010-04-
```

```
02T00:17:44Z
536152               1               103             102  2013-07-
16T20:00:36Z
536153               8             40533            6687  2010-06-
07T00:49:38Z
536156               0               105             212  2019-06-
20T03:09:41Z

        verified lang  lang_count
8          False   en      302604
10          True   en      302604
13          True   en      302604
14          True   en      302604
16         False   en      302604
...          ...  ...         ...
536149     False   en      302604
536151     False   en      302604
536152     False   en      302604
536153     False   en      302604
536156     False   en      302604

[302604 rows x 16 columns]
```

1.3.1.2: Drop Duplicates in English Tweets

We do not want duplicated tweets within our dataset, as if the same message is used multiple times, then their sentiment would be the same and that could skew and alter our overall sentiment that the model would be trained on.

```
# Drops duplicates in english_tweets_df, keeping the first one so as
not to delete all instances of the tweet!
english_tweets_df.drop_duplicates(keep='first', inplace=True)
```

```
/usr/local/lib/python3.8/dist-packages/pandas/util/_decorators.py:311:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return func(*args, **kwargs)
```

1.3.1.3: Investigation of Source Column and Null Values Within It

Looking at an updated summary of the English tweets dataset, we see that there are some null values within the column, the only column that applies to. Thus, we want to investigate the source column and decide how to deal with those rows, along with what to do with the source columns in general.

```
# Finds a summary of english_tweets_df -- note the null values within
the source column.
english_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 302581 entries, 8 to 536156
Data columns (total 16 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   status_id           302581 non-null  int64
 1   user_id             302581 non-null  int64
 2   created_at          302581 non-null  object
 3   screen_name         302581 non-null  object
 4   text                302581 non-null  object
 5   source              302579 non-null  object
 6   is_quote            302581 non-null  bool
 7   is_retweet          302581 non-null  bool
 8   favourites_count    302581 non-null  int64
 9   retweet_count       302581 non-null  int64
 10  followers_count     302581 non-null  int64
 11  friends_count       302581 non-null  int64
 12  account_created_at  302581 non-null  object
 13  verified            302581 non-null  bool
 14  lang                302581 non-null  object
 15  lang_count          302581 non-null  int64
dtypes: bool(3), int64(7), object(6)
memory usage: 33.2+ MB
```

```
# Finds the rows that are null in the source column.
english_tweets_df[english_tweets_df['source'].isna()]
```

```
                  status_id              user_id            created_at
\
85099   1245949719609655296  1241723414672322561  2020-04-03T05:42:16Z

249811  1246085612509081600  1241723414672322561  2020-04-03T14:42:15Z


          screen_name
text  \
85099   IndiaCOVID_19  COVID-19 India Update :\nTotal Active
Cases : ...
249811  IndiaCOVID_19  COVID-19 India Update :\nTotal Active
Cases : ...

       source  is_quote  is_retweet  favourites_count
retweet_count  \
85099     NaN     False       False                 0              0

249811    NaN     False       False                 0              0
```

```
        followers_count  friends_count    account_created_at  verified
lang  \
85099                55              1  2020-03-22T13:48:53Z     False
en
249811               60              1  2020-03-22T13:48:53Z     False
en

        lang_count
85099       302604
249811      302604
```

```
# Looks at the rows that aren't null in the source column for
comparison.
english_tweets_df[~(english_tweets_df['source'].isna())]
```

```
                  status_id              user_id            created_at
\
8       1245863586892636161             36969470  2020-04-03T00:00:00Z

10      1245863584447463425             36327407  2020-04-03T00:00:00Z

13      1245863586989248512            243811680  2020-04-03T00:00:00Z

14      1245863585227714560             44728980  2020-04-03T00:00:00Z

16      1245863584355307522            171548670  2020-04-03T00:00:00Z

...                     ...                  ...                   ...

536149  1246225964729679872  1246079496635043845  2020-04-03T23:59:58Z

536151  1246225965518372865            128713921  2020-04-03T23:59:58Z

536152  1246225970887081984           1599189781  2020-04-03T23:59:59Z

536153  1246225970433941505            152835605  2020-04-03T23:59:59Z

536156  1246225969586855936  1141543616453779458  2020-04-03T23:59:59Z


            screen_name
text  \
8          BostonReview  "The constantly shifting nature of a public
he...
10             htTweets  Iran parliament speaker tests positive for
#CO...
13           CBS4Local  “Domestic violence did not start during
#COVID...
14           ANCALERTS  The shows must go on!: Lloyd Webber musicals
t...
```

```
16          RadioNLNews  BC's Medical Health Officer says the province
...
...                  ...
...
536149      RoseTacazon  "These days of pain and sadness are bringing
m...
536151       XopherKyle  Did #POTUS just lie that many states operate
D...
536152           SylJud  Last day of @GarrisonMillES Spirit Week. It
ma...
536153   thinkingautism  This is a real concern for our USian
community...
536156  YouIsWrongAgain  @nytimes Oh looky. now the #coronavirus is
rac...

                     source  is_quote  is_retweet  favourites_count  \
8                    Buffer     False       False             16531
10                TweetDeck     False       False              2975
13                TweetDeck     False       False               495
14                TweetDeck     False       False              5450
16                TweetDeck     False       False               500
...                     ...       ...         ...                ...
536149  Twitter for Android     False       False                 4
536151   Twitter for iPhone     False       False             10978
536152   Twitter for iPhone     False       False               699
536153            TweetDeck     False       False              9373
536156   Twitter for iPhone     False       False              4633

        retweet_count  followers_count  friends_count
account_created_at  \
8                   0            42537           2761  2009-05-
01T15:42:57Z
10                  4          7249747            125  2009-04-
29T10:11:34Z
13                  0            13438            510  2011-01-
27T21:33:02Z
14                  6          4889749            776  2009-06-
04T21:26:24Z
16                  0             6934           2137  2010-07-
27T16:17:02Z
...               ...              ...            ...
...
536149              0                0             29  2020-04-
03T14:18:07Z
536151              1             5531           5255  2010-04-
02T00:17:44Z
536152              1              103            102  2013-07-
16T20:00:36Z
536153              8            40533           6687  2010-06-
```

```
07T00:49:38Z
536156              0              105             212  2019-06-
20T03:09:41Z

        verified lang  lang_count
8          False   en      302604
10          True   en      302604
13          True   en      302604
14          True   en      302604
16         False   en      302604
...          ...  ...         ...
536149     False   en      302604
536151     False   en      302604
536152     False   en      302604
536153     False   en      302604
536156     False   en      302604

[302579 rows x 16 columns]
```

```
# Counting unique values within the source column
n = len(pd.unique(english_tweets_df['source']))

print("Number of unique values :", n)
```

```
Number of unique values : 845
```

After analyzing the source column, we see that there are a large number of different possible values. Similar to HW4's justification, there are too many values to properly use it as a feature. Furthermore, the source of the tweet isn't particularly important for sentiment analysis, nor is it an avenue of possible exploration that we want to investigate, so the column can be dropped.

```
english_tweets_df.drop(['source'], axis=1, inplace=True)
english_tweets_df
```

```
/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py:4906:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return super().drop(

                  status_id              user_id            created_at
\
8       1245863586892636161             36969470  2020-04-03T00:00:00Z

10      1245863584447463425             36327407  2020-04-03T00:00:00Z

13      1245863586989248512            243811680  2020-04-03T00:00:00Z
```

```
14      1245863585227714560             44728980  2020-04-03T00:00:00Z
16      1245863584355307522            171548670  2020-04-03T00:00:00Z
...                      ...                  ...                   ...
536149  1246225964729679872  1246079496635043845  2020-04-03T23:59:58Z
536151  1246225965518372865             128713921  2020-04-03T23:59:58Z
536152  1246225970887081984            1599189781  2020-04-03T23:59:59Z
536153  1246225970433941505             152835605  2020-04-03T23:59:59Z
536156  1246225969586855936  1141543616453779458  2020-04-03T23:59:59Z

             screen_name
text  \
8          BostonReview  "The constantly shifting nature of a public
he...
10             htTweets  Iran parliament speaker tests positive for
#CO...
13            CBS4Local  “Domestic violence did not start during
#COVID...
14            ANCALERTS  The shows must go on!: Lloyd Webber musicals
t...
16          RadioNLNews  BC's Medical Health Officer says the province
...
...                  ...
...
536149      RoseTacazon  "These days of pain and sadness are bringing
m...
536151       XopherKyle  Did #POTUS just lie that many states operate
D...
536152           SylJud  Last day of @GarrisonMillES Spirit Week. It
ma...
536153   thinkingautism  This is a real concern for our USian
community...
536156  YouIsWrongAgain  @nytimes Oh looky. now the #coronavirus is
rac...

        is_quote  is_retweet  favourites_count  retweet_count  \
8          False       False             16531              0
10         False       False              2975              4
13         False       False               495              0
14         False       False              5450              6
16         False       False               500              0
...          ...         ...               ...            ...
```

```
536149     False       False                4              0
536151     False       False            10978              1
536152     False       False              699              1
536153     False       False             9373              8
536156     False       False             4633              0

        followers_count  friends_count    account_created_at  verified
lang  \
8                 42537           2761  2009-05-01T15:42:57Z     False
en
10              7249747            125  2009-04-29T10:11:34Z      True
en
13                13438            510  2011-01-27T21:33:02Z      True
en
14              4889749            776  2009-06-04T21:26:24Z      True
en
16                 6934           2137  2010-07-27T16:17:02Z     False
en
...                 ...            ...                   ...       ...
...
536149               0             29  2020-04-03T14:18:07Z     False
en
536151            5531           5255  2010-04-02T00:17:44Z     False
en
536152             103            102  2013-07-16T20:00:36Z     False
en
536153           40533           6687  2010-06-07T00:49:38Z     False
en
536156             105            212  2019-06-20T03:09:41Z     False
en

        lang_count
8           302604
10          302604
13          302604
14          302604
16          302604
...            ...
536149      302604
536151      302604
536152      302604
536153      302604
536156      302604

[302581 rows x 15 columns]
```

1.3.1.4: Investigation of Verified and Follower Count Columns As An Avenue of Exploration

One possible avenue of sentiment analysis that could be interesting to look at is the discrepancy between how the overall sentiment is for users who are verified versus users who are not verified.

Similarly, it could be interesting to look at the difference between overall sentiment of users with high numbers of followers versus low numbers of followers. For the purpose of our analysis, we will use this article as a source for classification between high and low numbers of followers: https://www.key4biz.it/files/000270/00027033.pdf

Looking at the data, it is interesting to note that the difference between the number of high-follower people and low-follower people is smaller than what the article indicates. This could be a reflection of a bias present in the selection of the tweets for the dataset, as it makes sense that users who have high amounts of followers are those who post more and post more about relevant topics (like COVID-19). Thus, this could be an explanation for the trends seen in the comparison bar plots.

```
# Creates a new dataframe of the number of verified vs non-verified
users
english_verified_df =
english_tweets_df['verified'].value_counts().reset_index()
english_verified_df
```

```
   index  verified
0  False    269665
1   True     32916
```

```
# Creates a user-defined function for assigning "high follower count"
vs "low follower count" based on article definition
def follower_classification(content):
  return 1 if content >= 500 else 0

# Applies this function to the followers_count column of
english_tweets_df
english_tweets_df['follower_classification'] =
english_tweets_df['followers_count'].apply(lambda x:
follower_classification(x))
english_tweets_df
```

```
<ipython-input-46-3b08d21efac6>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  english_tweets_df['follower_classification'] =
english_tweets_df['followers_count'].apply(lambda x:
follower_classification(x))
```

```
                  status_id              user_id           created_at
\
8       1245863586892636161             36969470  2020-04-03T00:00:00Z

10      1245863584447463425             36327407  2020-04-03T00:00:00Z

13      1245863586989248512            243811680  2020-04-03T00:00:00Z

14      1245863585227714560             44728980  2020-04-03T00:00:00Z

16      1245863584355307522            171548670  2020-04-03T00:00:00Z

...                     ...                  ...                   ...

536149  1246225964729679872  1246079496635043845  2020-04-03T23:59:58Z

536151  1246225965518372865            128713921  2020-04-03T23:59:58Z

536152  1246225970887081984           1599189781  2020-04-03T23:59:59Z

536153  1246225970433941505            152835605  2020-04-03T23:59:59Z

536156  1246225969586855936  1141543616453779458  2020-04-03T23:59:59Z


             screen_name
text  \
8          BostonReview  "The constantly shifting nature of a public
he...
10             htTweets  Iran parliament speaker tests positive for
#CO...
13            CBS4Local  “Domestic violence did not start during
#COVID...
14            ANCALERTS  The shows must go on!: Lloyd Webber musicals
t...
16         RadioNLNews  BC's Medical Health Officer says the province
...
...                  ...
...
536149      RoseTacazon  "These days of pain and sadness are bringing
m...
536151       XopherKyle  Did #POTUS just lie that many states operate
D...
536152           SylJud  Last day of @GarrisonMillES Spirit Week. It
ma...
536153   thinkingautism  This is a real concern for our USian
community...
536156  YouIsWrongAgain  @nytimes Oh looky. now the #coronavirus is
rac...

        is_quote  is_retweet  favourites_count  retweet_count  \
```

```
8          False       False              16531               0
10         False       False               2975               4
13         False       False                495               0
14         False       False               5450               6
16         False       False                500               0
...          ...         ...                ...             ...
536149     False       False                  4               0
536151     False       False              10978               1
536152     False       False                699               1
536153     False       False               9373               8
536156     False       False               4633               0

        followers_count  friends_count    account_created_at  verified
lang  \
8                 42537           2761  2009-05-01T15:42:57Z     False
en
10              7249747            125  2009-04-29T10:11:34Z      True
en
13                13438            510  2011-01-27T21:33:02Z      True
en
14              4889749            776  2009-06-04T21:26:24Z      True
en
16                 6934           2137  2010-07-27T16:17:02Z     False
en
...                 ...            ...                   ...       ...
...
536149               0             29  2020-04-03T14:18:07Z     False
en
536151            5531           5255  2010-04-02T00:17:44Z     False
en
536152             103            102  2013-07-16T20:00:36Z     False
en
536153           40533           6687  2010-06-07T00:49:38Z     False
en
536156             105            212  2019-06-20T03:09:41Z     False
en

        lang_count  follower_classification
8           302604                        1
10          302604                        1
13          302604                        1
14          302604                        1
16          302604                        1
...            ...                      ...
536149      302604                        0
536151      302604                        1
536152      302604                        0
536153      302604                        1
536156      302604                        0
```

```
[302581 rows x 16 columns]
```

```
# Inside of verified column, sets value to 1 if verified and 0
otherwise
english_tweets_df['verified'] =
english_tweets_df['verified'].apply(lambda x: 0 if x == False else 1)
english_tweets_df
```

```
<ipython-input-47-5a76b25986d8>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  english_tweets_df['verified'] =
english_tweets_df['verified'].apply(lambda x: 0 if x == False else 1)
```

```
                  status_id              user_id            created_at
\
8       1245863586892636161              36969470  2020-04-03T00:00:00Z

10      1245863584447463425              36327407  2020-04-03T00:00:00Z

13      1245863586989248512             243811680  2020-04-03T00:00:00Z

14      1245863585227714560              44728980  2020-04-03T00:00:00Z

16      1245863584355307522             171548670  2020-04-03T00:00:00Z

...                      ...                   ...                   ...

536149  1246225964729679872  1246079496635043845  2020-04-03T23:59:58Z

536151  1246225965518372865             128713921  2020-04-03T23:59:58Z

536152  1246225970887081984            1599189781  2020-04-03T23:59:59Z

536153  1246225970433941505             152835605  2020-04-03T23:59:59Z

536156  1246225969586855936  1141543616453779458  2020-04-03T23:59:59Z


            screen_name
text  \
8          BostonReview  "The constantly shifting nature of a public
he...
10             htTweets  Iran parliament speaker tests positive for
#CO...
13            CBS4Local  “Domestic violence did not start during
```

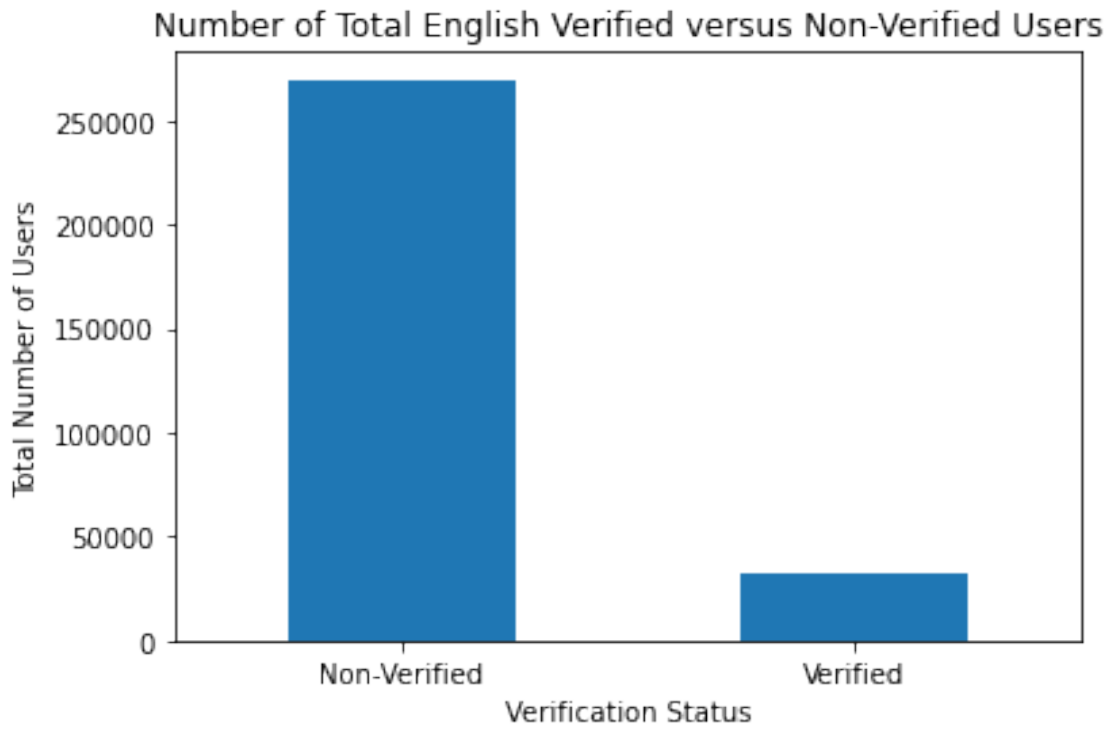
```
#COVID...
14             ANCALERTS  The shows must go on!: Lloyd Webber musicals
t...
16          RadioNLNews  BC's Medical Health Officer says the province
...
...                  ...
...
536149      RoseTacazon  "These days of pain and sadness are bringing
m...
536151       XopherKyle  Did #POTUS just lie that many states operate
D...
536152           SylJud  Last day of @GarrisonMillES Spirit Week. It
ma...
536153   thinkingautism  This is a real concern for our USian
community...
536156  YouIsWrongAgain  @nytimes Oh looky. now the #coronavirus is
rac...

        is_quote  is_retweet  favourites_count  retweet_count  \
8          False       False             16531              0
10         False       False              2975              4
13         False       False               495              0
14         False       False              5450              6
16         False       False               500              0
...          ...         ...               ...            ...
536149     False       False                 4              0
536151     False       False             10978              1
536152     False       False               699              1
536153     False       False              9373              8
536156     False       False              4633              0

        followers_count  friends_count    account_created_at  verified
lang  \
8                 42537           2761  2009-05-01T15:42:57Z         0
en
10              7249747            125  2009-04-29T10:11:34Z         1
en
13                13438            510  2011-01-27T21:33:02Z         1
en
14              4889749            776  2009-06-04T21:26:24Z         1
en
16                 6934           2137  2010-07-27T16:17:02Z         0
en
...                 ...            ...                   ...       ...
...
536149               0             29  2020-04-03T14:18:07Z         0
en
536151            5531           5255  2010-04-02T00:17:44Z         0
en
```

```
536152               103             102  2013-07-16T20:00:36Z          0
en
536153             40533            6687  2010-06-07T00:49:38Z          0
en
536156               105             212  2019-06-20T03:09:41Z          0
en

        lang_count  follower_classification
8           302604                        1
10          302604                        1
13          302604                        1
14          302604                        1
16          302604                        1
...            ...                      ...
536149      302604                        0
536151      302604                        1
536152      302604                        0
536153      302604                        1
536156      302604                        0

[302581 rows x 16 columns]
```

```
# Creates a new dataframe of the number of high follower vs low
follower users
english_follower_df =
english_tweets_df['follower_classification'].value_counts().reset_inde
x()
english_follower_df
```

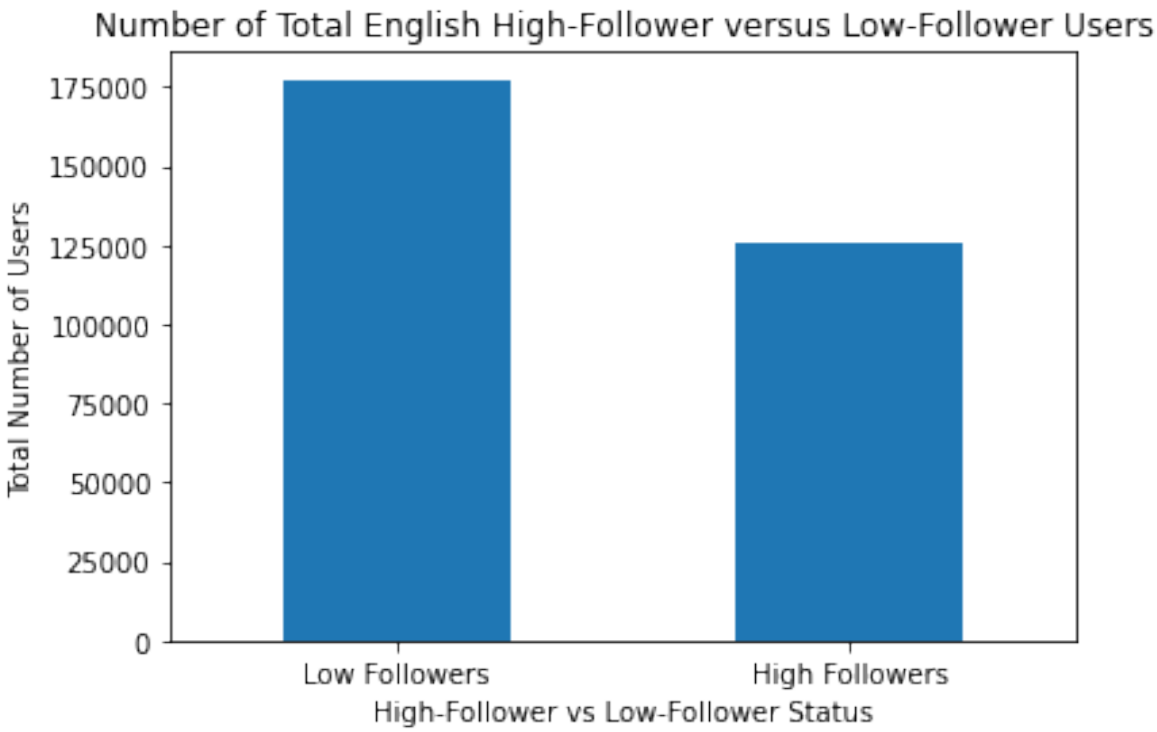
```
   index  follower_classification
0      1                    177214
1      0                    125367
```

```
# Plots number of verified vs non-verified tweets
english_verified_df.plot.bar(x='index', legend=None)
plt.title("Number of Total English Verified versus Non-Verified
Users")
plt.ylabel("Total Number of Users")
plt.xlabel("Verification Status")

bars = ('Non-Verified', 'Verified')
x_pos = np.arange(len(bars))

plt.xticks(x_pos, bars)
plt.xticks(rotation=0)
plt.show()
```



This bar graph shows the large difference between the number of non-verified and verified users -- this makes sense, as naturally there would be less people that are verified, and both classifications still have large sample sizes to try and run analysis on.

```
# Plots number of high-follower vs low-follower tweets
english_follower_df.plot.bar(x='index', legend=None)
plt.title("Number of Total English High-Follower versus Low-Follower
Users")
plt.ylabel("Total Number of Users")
plt.xlabel("High-Follower vs Low-Follower Status")

bars = ('Low Followers', 'High Followers')
x_pos = np.arange(len(bars))

plt.xticks(x_pos, bars)
plt.xticks(rotation=0)
plt.show()
```



This bar graph shows the difference between the number of high-follower users and low-follower users. This is actually surprising, as given the article it indicated that "high-follower" users were only ~1-2% of the total Twitter userbase. However, the mass majority of Twitter accounts most likely do not post at all, so if a working usable Kaggle dataset was to be compiled, then it is understandable to have an overrepresentation of accounts that post often, and therefore high-follower accounts by association.

1.3.1.5: Cleaning English Tweets Time Stamps

Looking at the created_at and account_created_at columns, we notice that they need to be converted to proper datetime formatting, as right now they're in an improper format. Thus, we reformat them and transform them to datetime format.

```
# Cleans created_at column and converts it to datetime format
english_tweets_df['created_at'] =
english_tweets_df['created_at'].apply(lambda x: x.split('T')[0])
pd.to_datetime(english_tweets_df['created_at'])
english_tweets_df
```

```
<ipython-input-51-a333ed73ec50>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
```

```
  english_tweets_df['created_at'] =
english_tweets_df['created_at'].apply(lambda x: x.split('T')[0])

                  status_id              user_id  created_at
screen_name  \
8       1245863586892636161             36969470  2020-04-03
BostonReview
10      1245863584447463425             36327407  2020-04-03
htTweets
13      1245863586989248512            243811680  2020-04-03
CBS4Local
14      1245863585227714560             44728980  2020-04-03
ANCALERTS
16      1245863584355307522            171548670  2020-04-03
RadioNLNews
...                     ...                  ...         ...
...
536149  1246225964729679872  1246079496635043845  2020-04-03
RoseTacazon
536151  1246225965518372865             128713921  2020-04-03
XopherKyle
536152  1246225970887081984            1599189781  2020-04-03
SylJud
536153  1246225970433941505             152835605  2020-04-03
thinkingautism
536156  1246225969586855936  1141543616453779458  2020-04-03
YouIsWrongAgain

                                                     text  is_quote  \
8       "The constantly shifting nature of a public he...     False
10      Iran parliament speaker tests positive for #CO...     False
13      “Domestic violence did not start during #COVID...     False
14      The shows must go on!: Lloyd Webber musicals t...     False
16      BC's Medical Health Officer says the province ...     False
...                                                   ...       ...
536149  "These days of pain and sadness are bringing m...     False
536151  Did #POTUS just lie that many states operate D...     False
536152  Last day of @GarrisonMillES Spirit Week. It ma...     False
536153  This is a real concern for our USian community...     False
536156  @nytimes Oh looky. now the #coronavirus is rac...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
8            False             16531              0            42537

10           False              2975              4          7249747

13           False               495              0            13438

14           False              5450              6          4889749
```

```
16           False                500              0             6934
...            ...                ...            ...              ...
536149       False                  4              0                0
536151       False              10978              1             5531
536152       False                699              1              103
536153       False               9373              8            40533
536156       False               4633              0              105

        friends_count    account_created_at  verified lang  lang_count
\
8                2761  2009-05-01T15:42:57Z         0   en      302604
10                125  2009-04-29T10:11:34Z         1   en      302604
13                510  2011-01-27T21:33:02Z         1   en      302604
14                776  2009-06-04T21:26:24Z         1   en      302604
16               2137  2010-07-27T16:17:02Z         0   en      302604
...               ...                   ...       ...  ...         ...
536149             29  2020-04-03T14:18:07Z         0   en      302604
536151           5255  2010-04-02T00:17:44Z         0   en      302604
536152            102  2013-07-16T20:00:36Z         0   en      302604
536153           6687  2010-06-07T00:49:38Z         0   en      302604
536156            212  2019-06-20T03:09:41Z         0   en      302604

        follower_classification
8                             1
10                            1
13                            1
14                            1
16                            1
...                         ...
536149                        0
536151                        1
536152                        0
536153                        1
```

```
536156                          0

[302581 rows x 16 columns]
```

```
english_tweets_df.dtypes
```

```
status_id                   int64
user_id                     int64
created_at                 object
screen_name                object
text                       object
is_quote                     bool
is_retweet                   bool
favourites_count            int64
retweet_count               int64
followers_count             int64
friends_count               int64
account_created_at         object
verified                    int64
lang                       object
lang_count                  int64
follower_classification     int64
dtype: object
```

```
# Cleans account_created_at column and converts it to datetime format
english_tweets_df['account_created_at'] =
english_tweets_df['account_created_at'].apply(lambda x: x.split('T')
[0])
pd.to_datetime(english_tweets_df['account_created_at'])
english_tweets_df
```

```
<ipython-input-53-a5873b9d2712>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  english_tweets_df['account_created_at'] =
english_tweets_df['account_created_at'].apply(lambda x: x.split('T')
[0])
```

```
                  status_id              user_id  created_at
screen_name  \
8       1245863586892636161             36969470  2020-04-03
BostonReview
10      1245863584447463425             36327407  2020-04-03
htTweets
13      1245863586989248512            243811680  2020-04-03
CBS4Local
14      1245863585227714560             44728980  2020-04-03
```

```
ANCALERTS
16      1245863584355307522            171548670  2020-04-03
RadioNLNews
...                      ...                  ...         ...
...
536149  1246225964729679872  1246079496635043845  2020-04-03
RoseTacazon
536151  1246225965518372865            128713921  2020-04-03
XopherKyle
536152  1246225970887081984           1599189781  2020-04-03
SylJud
536153  1246225970433941505            152835605  2020-04-03
thinkingautism
536156  1246225969586855936  1141543616453779458  2020-04-03
YouIsWrongAgain

                                                    text  is_quote  \
8       "The constantly shifting nature of a public he...     False
10      Iran parliament speaker tests positive for #CO...     False
13      “Domestic violence did not start during #COVID...     False
14      The shows must go on!: Lloyd Webber musicals t...     False
16      BC's Medical Health Officer says the province ...     False
...                                                   ...       ...
536149  "These days of pain and sadness are bringing m...     False
536151  Did #POTUS just lie that many states operate D...     False
536152  Last day of @GarrisonMillES Spirit Week. It ma...     False
536153  This is a real concern for our USian community...     False
536156  @nytimes Oh looky. now the #coronavirus is rac...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
8            False             16531              0            42537

10           False              2975              4          7249747

13           False               495              0            13438

14           False              5450              6          4889749

16           False               500              0             6934

...            ...               ...            ...              ...

536149       False                 4              0                0

536151       False             10978              1             5531

536152       False               699              1              103

536153       False              9373              8            40533
```

```
536156       False               4633               0               105


        friends_count account_created_at  verified lang  lang_count  \
8                2761         2009-05-01         0   en      302604
10                125         2009-04-29         1   en      302604
13                510         2011-01-27         1   en      302604
14                776         2009-06-04         1   en      302604
16               2137         2010-07-27         0   en      302604
...               ...                ...       ...  ...         ...
536149            29         2020-04-03         0   en      302604
536151          5255         2010-04-02         0   en      302604
536152           102         2013-07-16         0   en      302604
536153          6687         2010-06-07         0   en      302604
536156           212         2019-06-20         0   en      302604

        follower_classification
8                             1
10                            1
13                            1
14                            1
16                            1
...                         ...
536149                        0
536151                        1
536152                        0
536153                        1
536156                        0

[302581 rows x 16 columns]
```

1.3.1.6: Investigation of English Tweets Retweets and Quotes and Removing Unnecessary Columns

We only want tweets that are that individual's personal sentiment, rather than quoting another user's opinion, as including elements such as retweets and quotes could include multiple users that are retweeting and/or quoting the same tweet, thus again skewing overall sentiment for training.

Thus, as long as they don't account for too many tweets, we should remove all rows that are quotes or tweets.

Similarly, we should remove any columns that are not necessary for overall sentiment analysis, such as personal identifiers.

```
# Checking number of tweets that are retweets
# Seeing that there are 0, we can remove this column easily.
english_tweets_df[english_tweets_df['is_retweet'] == True].count()
```

```
status_id                   0
user_id                     0
created_at                  0
screen_name                 0
text                        0
is_quote                    0
is_retweet                  0
favourites_count            0
retweet_count               0
followers_count             0
friends_count               0
account_created_at          0
verified                    0
lang                        0
lang_count                  0
follower_classification     0
dtype: int64
```

```
# Checking number of tweets that are quotes.
# Seeing that there's a small proportion of tweets that are quotes,
they can be removed without overly affecting sentiment analysis.
english_tweets_df[english_tweets_df['is_quote'] == True].count()
```

```
status_id                   35655
user_id                     35655
created_at                  35655
screen_name                 35655
text                        35655
is_quote                    35655
is_retweet                  35655
favourites_count            35655
retweet_count               35655
followers_count             35655
friends_count               35655
account_created_at          35655
verified                    35655
lang                        35655
lang_count                  35655
follower_classification     35655
dtype: int64
```

```
# Removes all rows from the dataset that are quotes
english_tweets_df = english_tweets_df[english_tweets_df['is_quote'] ==
False]
english_tweets_df
```

```
                  status_id              user_id  created_at
screen_name  \
8       1245863586892636161             36969470  2020-04-03
BostonReview
```

```
10      1245863584447463425              36327407  2020-04-03
htTweets
13      1245863586989248512             243811680  2020-04-03
CBS4Local
14      1245863585227714560              44728980  2020-04-03
ANCALERTS
16      1245863584355307522             171548670  2020-04-03
RadioNLNews
...                      ...                  ...         ...
...
536149  1246225964729679872  1246079496635043845  2020-04-03
RoseTacazon
536151  1246225965518372865             128713921  2020-04-03
XopherKyle
536152  1246225970887081984            1599189781  2020-04-03
SylJud
536153  1246225970433941505             152835605  2020-04-03
thinkingautism
536156  1246225969586855936  1141543616453779458  2020-04-03
YouIsWrongAgain

                                                     text  is_quote  \
8       "The constantly shifting nature of a public he...     False
10      Iran parliament speaker tests positive for #CO...     False
13      “Domestic violence did not start during #COVID...     False
14      The shows must go on!: Lloyd Webber musicals t...     False
16      BC's Medical Health Officer says the province ...     False
...                                                   ...       ...
536149  "These days of pain and sadness are bringing m...     False
536151  Did #POTUS just lie that many states operate D...     False
536152  Last day of @GarrisonMillES Spirit Week. It ma...     False
536153  This is a real concern for our USian community...     False
536156  @nytimes Oh looky. now the #coronavirus is rac...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
8            False             16531              0            42537

10           False              2975              4          7249747

13           False               495              0            13438

14           False              5450              6          4889749

16           False               500              0             6934

...            ...                ...            ...              ...

536149       False                 4              0                0
```

```
536151       False              10978               1              5531

536152       False                699               1               103

536153       False               9373               8             40533

536156       False               4633               0               105


        friends_count account_created_at  verified lang  lang_count  \
8                2761         2009-05-01         0   en      302604
10                125         2009-04-29         1   en      302604
13                510         2011-01-27         1   en      302604
14                776         2009-06-04         1   en      302604
16               2137         2010-07-27         0   en      302604
...               ...                ...       ...  ...         ...
536149             29         2020-04-03         0   en      302604
536151           5255         2010-04-02         0   en      302604
536152            102         2013-07-16         0   en      302604
536153           6687         2010-06-07         0   en      302604
536156            212         2019-06-20         0   en      302604

        follower_classification
8                             1
10                            1
13                            1
14                            1
16                            1
...                         ...
536149                        0
536151                        1
536152                        0
536153                        1
536156                        0

[266926 rows x 16 columns]
```

```
# Removes all rows from the dataset that are retweets
english_tweets_df = english_tweets_df[english_tweets_df['is_retweet']
== False]
english_tweets_df
```

```
                   status_id              user_id  created_at
screen_name  \
8       1245863586892636161             36969470  2020-04-03
BostonReview
10      1245863584447463425             36327407  2020-04-03
htTweets
13      1245863586989248512            243811680  2020-04-03
CBS4Local
```

```
14      1245863585227714560             44728980  2020-04-03
ANCALERTS
16      1245863584355307522            171548670  2020-04-03
RadioNLNews
...                      ...                  ...         ...
...
536149  1246225964729679872  1246079496635043845  2020-04-03
RoseTacazon
536151  1246225965518372865            128713921  2020-04-03
XopherKyle
536152  1246225970887081984           1599189781  2020-04-03
SylJud
536153  1246225970433941505            152835605  2020-04-03
thinkingautism
536156  1246225969586855936  1141543616453779458  2020-04-03
YouIsWrongAgain

                                                    text  is_quote  \
8       "The constantly shifting nature of a public he...     False
10      Iran parliament speaker tests positive for #CO...     False
13      “Domestic violence did not start during #COVID...     False
14      The shows must go on!: Lloyd Webber musicals t...     False
16      BC's Medical Health Officer says the province ...     False
...                                                   ...       ...
536149  "These days of pain and sadness are bringing m...     False
536151  Did #POTUS just lie that many states operate D...     False
536152  Last day of @GarrisonMillES Spirit Week. It ma...     False
536153  This is a real concern for our USian community...     False
536156  @nytimes Oh looky. now the #coronavirus is rac...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
8            False              16531              0            42537

10           False               2975              4          7249747

13           False                495              0            13438

14           False               5450              6          4889749

16           False                500              0             6934

...            ...                ...            ...              ...

536149       False                  4              0                0

536151       False              10978              1             5531

536152       False                699              1              103
```

```
536153       False               9373               8             40533

536156       False               4633               0               105


        friends_count account_created_at  verified lang  lang_count  \
8                2761         2009-05-01         0   en      302604
10                125         2009-04-29         1   en      302604
13                510         2011-01-27         1   en      302604
14                776         2009-06-04         1   en      302604
16               2137         2010-07-27         0   en      302604
...               ...                ...       ...  ...         ...
536149            29         2020-04-03         0   en      302604
536151          5255         2010-04-02         0   en      302604
536152           102         2013-07-16         0   en      302604
536153          6687         2010-06-07         0   en      302604
536156           212         2019-06-20         0   en      302604

        follower_classification
8                             1
10                            1
13                            1
14                            1
16                            1
...                         ...
536149                        0
536151                        1
536152                        0
536153                        1
536156                        0

[266926 rows x 16 columns]
```

```
# As they're not necessary anymore, we can drop the quote and retweet
columns now
english_tweets_df.drop(['is_quote', 'is_retweet'], axis=1,
inplace=True)
english_tweets_df
```

```
                  status_id              user_id  created_at
screen_name  \
8       1245863586892636161             36969470  2020-04-03
BostonReview
10      1245863584447463425             36327407  2020-04-03
htTweets
13      1245863586989248512            243811680  2020-04-03
CBS4Local
14      1245863585227714560             44728980  2020-04-03
ANCALERTS
16      1245863584355307522            171548670  2020-04-03
```

```
RadioNLNews
...                      ...                    ...          ...
...
536149  1246225964729679872  1246079496635043845  2020-04-03
RoseTacazon
536151  1246225965518372865            128713921  2020-04-03
XopherKyle
536152  1246225970887081984           1599189781  2020-04-03
SylJud
536153  1246225970433941505            152835605  2020-04-03
thinkingautism
536156  1246225969586855936  1141543616453779458  2020-04-03
YouIsWrongAgain

                                                      text
favourites_count  \
8       "The constantly shifting nature of a public he...
16531
10      Iran parliament speaker tests positive for #CO...
2975
13      “Domestic violence did not start during #COVID...
495
14      The shows must go on!: Lloyd Webber musicals t...
5450
16      BC's Medical Health Officer says the province ...
500
...                                                   ...
...
536149  "These days of pain and sadness are bringing m...
4
536151  Did #POTUS just lie that many states operate D...
10978
536152  Last day of @GarrisonMillES Spirit Week. It ma...
699
536153  This is a real concern for our USian community...
9373
536156  @nytimes Oh looky. now the #coronavirus is rac...
4633

        retweet_count  followers_count  friends_count
account_created_at  \
8                   0            42537           2761         2009-05-
01
10                  4          7249747            125         2009-04-
29
13                  0            13438            510         2011-01-
27
14                  6          4889749            776         2009-06-
04
```

```
16                  0             6934           2137          2010-07-
27
...               ...              ...            ...                .
..
536149              0                0             29          2020-04-
03
536151              1             5531           5255          2010-04-
02
536152              1              103            102          2013-07-
16
536153              8            40533           6687          2010-06-
07
536156              0              105            212          2019-06-
20

        verified lang  lang_count  follower_classification
8              0   en      302604                        1
10             1   en      302604                        1
13             1   en      302604                        1
14             1   en      302604                        1
16             0   en      302604                        1
...          ...  ...         ...                      ...
536149         0   en      302604                        0
536151         0   en      302604                        1
536152         0   en      302604                        0
536153         0   en      302604                        1
536156         0   en      302604                        0

[266926 rows x 14 columns]
```

```
# Looking at the status_id, user_id, and screen_name columns, we don't
need them for sentiment analysis, so we can drop them.
english_tweets_df.drop(['status_id', 'user_id', 'screen_name'],
axis=1, inplace=True)
english_tweets_df
```

```
        created_at                                              text
\
8       2020-04-03  "The constantly shifting nature of a public he...

10      2020-04-03  Iran parliament speaker tests positive for #CO...

13      2020-04-03  “Domestic violence did not start during #COVID...

14      2020-04-03  The shows must go on!: Lloyd Webber musicals t...

16      2020-04-03  BC's Medical Health Officer says the province ...

...            ...                                               ...
```

```
536149  2020-04-03  "These days of pain and sadness are bringing m...
536151  2020-04-03  Did #POTUS just lie that many states operate D...
536152  2020-04-03  Last day of @GarrisonMillES Spirit Week. It ma...
536153  2020-04-03  This is a real concern for our USian community...
536156  2020-04-03  @nytimes Oh looky. now the #coronavirus is rac...

        favourites_count  retweet_count  followers_count
friends_count  \
8                  16531              0            42537
2761
10                  2975              4          7249747
125
13                   495              0            13438
510
14                  5450              6          4889749
776
16                   500              0             6934
2137
...                  ...            ...              ...             ..
.
536149                 4              0                0
29
536151             10978              1             5531
5255
536152               699              1              103
102
536153              9373              8            40533
6687
536156              4633              0              105
212

       account_created_at  verified lang  lang_count
follower_classification
8              2009-05-01         0   en      302604
1
10             2009-04-29         1   en      302604
1
13             2011-01-27         1   en      302604
1
14             2009-06-04         1   en      302604
1
16             2010-07-27         0   en      302604
1
...                   ...       ...  ...         ...
...
```

```
536149         2020-04-03         0   en      302604
0
536151         2010-04-02         0   en      302604
1
536152         2013-07-16         0   en      302604
0
536153         2010-06-07         0   en      302604
1
536156         2019-06-20         0   en      302604
0

[266926 rows x 11 columns]
```

```
# Furthermore, as we know this is the English tweet dataset, and don't
need lang_count anymore, we can remove those as well
english_tweets_df.drop(['lang', 'lang_count'], axis=1, inplace=True)
english_tweets_df
```

```
        created_at                                               text
\
8       2020-04-03  "The constantly shifting nature of a public he...

10      2020-04-03  Iran parliament speaker tests positive for #CO...

13      2020-04-03  “Domestic violence did not start during #COVID...

14      2020-04-03  The shows must go on!: Lloyd Webber musicals t...

16      2020-04-03  BC's Medical Health Officer says the province ...

...            ...                                                ...

536149  2020-04-03  "These days of pain and sadness are bringing m...

536151  2020-04-03  Did #POTUS just lie that many states operate D...

536152  2020-04-03  Last day of @GarrisonMillES Spirit Week. It ma...

536153  2020-04-03  This is a real concern for our USian community...

536156  2020-04-03  @nytimes Oh looky. now the #coronavirus is rac...


        favourites_count  retweet_count  followers_count
friends_count  \
8                  16531              0            42537
2761
10                  2975              4          7249747
125
13                   495              0            13438
510
```

```
14                   5450               6          4889749
776
16                    500               0             6934
2137
...                   ...             ...              ...             ..
.
536149                  4               0                0
29
536151              10978               1             5531
5255
536152                699               1              103
102
536153                9373              8            40533
6687
536156                4633              0              105
212

       account_created_at  verified  follower_classification
8              2009-05-01         0                        1
10             2009-04-29         1                        1
13             2011-01-27         1                        1
14             2009-06-04         1                        1
16             2010-07-27         0                        1
...                   ...       ...                      ...
536149         2020-04-03         0                        0
536151         2010-04-02         0                        1
536152         2013-07-16         0                        0
536153         2010-06-07         0                        1
536156         2019-06-20         0                        0

[266926 rows x 9 columns]
```

1.3.2: English Tweets Sentiment Analysis

Now, we want to perform sentiment analysis on the English tweets, both to process the tweets in preparation for modeling and to gain a preliminary look at how the sentiment appears to be.

1.3.2.1: English Sentiment Analysis Preprocessing

Uses the NLTK library (https://www.nltk.org/), which is a toolkit for Natural Language Processing that has resources for multiple languages, including English.

As a natural pipeline for sentiment analysis, we have to first tokenize each tweet, before turning them into lowercase, stemming them with the SnowballStemmer library, and removing any non-alphabetic characters or "stopwords" according to the English library. These are all provided by the NLTK library.

The SnowballStemmer library is known as an improvement over the normal PorterStemmer library (https://towardsdatascience.com/stemming-lemmatization-what-ba782b7c0bd8), and

because it is included in the NLTK library and supports multi-lingual analysis, we have chosen to use that to increase the accuracy of our processed tweets.

```
# Imports the NLTK library for sentiment analysis and sentiment
analysis preprocessing
import nltk
nltk.download('punkt')
nltk.__version__
```

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

```
{"type":"string"}
```

```
# Imports the stopwords for the English language from the NLTK library
from nltk.corpus import stopwords
nltk.download('stopwords')
stopwords = set(stopwords.words('english'))
```

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

```
# Imports a English-language stemmer from the SnowballStemmer class
within NLTK
from nltk.stem.snowball import SnowballStemmer
english_snowball_stemmer = SnowballStemmer(language='english')

# A user-defined function that tokenizes, lowercases, stems, and
removes stopwords/non-alphabetic characters
# It then appends what remains together into a string and returns it.
@numba.jit
def tokenize_english_content(content):
  tokens = nltk.word_tokenize(content)
  final_string = []
  for tk in tokens:
    lowercase = tk.lower()
    stemmed_lowercase = english_snowball_stemmer.stem(lowercase)
    if ((str.isalpha(stemmed_lowercase)) and (stemmed_lowercase not in
stopwords)):
      final_string.append(stemmed_lowercase)
    else:
      continue

  return final_string

# Applies this tokenizing function to each tweet and puts it in a
processed_text column
english_tweets_df['processed_text'] =
english_tweets_df['text'].map(lambda x: tokenize_english_content(x))
english_tweets_df
```

```
<ipython-input-64-9bd8cc7ee7f8>:3: NumbaWarning:
Compilation is falling back to object mode WITH looplifting enabled
because Function "tokenize_english_content" failed type inference due
to: Untyped global name 'english_snowball_stemmer': Cannot determine
Numba type of <class 'nltk.stem.snowball.SnowballStemmer'>

File "<ipython-input-64-9bd8cc7ee7f8>", line 9:
def tokenize_english_content(content):
    <source elided>
    lowercase = tk.lower()
    stemmed_lowercase = english_snowball_stemmer.stem(lowercase)
    ^

  @numba.jit
<ipython-input-64-9bd8cc7ee7f8>:3: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "tokenize_english_content" failed type inference due
to: Cannot determine Numba type of <class
'numba.core.dispatcher.LiftedLoop'>

File "<ipython-input-64-9bd8cc7ee7f8>", line 7:
def tokenize_english_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "tokenize_english_content" was compiled
in object mode without forceobj=True, but has lifted loops.

File "<ipython-input-64-9bd8cc7ee7f8>", line 5:
def tokenize_english_content(content):
  tokens = nltk.word_tokenize(content)
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-64-9bd8cc7ee7f8>", line 5:
def tokenize_english_content(content):
  tokens = nltk.word_tokenize(content)
```

```
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,
<ipython-input-64-9bd8cc7ee7f8>:3: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "tokenize_english_content" failed type inference due
to: Untyped global name 'english_snowball_stemmer': Cannot determine
Numba type of <class 'nltk.stem.snowball.SnowballStemmer'>

File "<ipython-input-64-9bd8cc7ee7f8>", line 9:
def tokenize_english_content(content):
    <source elided>
    lowercase = tk.lower()
    stemmed_lowercase = english_snowball_stemmer.stem(lowercase)
    ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "tokenize_english_content" was compiled
in object mode without forceobj=True.

File "<ipython-input-64-9bd8cc7ee7f8>", line 7:
def tokenize_english_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-64-9bd8cc7ee7f8>", line 7:
def tokenize_english_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,

        created_at                                                  text
\
8       2020-04-03  "The constantly shifting nature of a public he...
```

```
10      2020-04-03  Iran parliament speaker tests positive for #CO...
13      2020-04-03  “Domestic violence did not start during #COVID...
14      2020-04-03  The shows must go on!: Lloyd Webber musicals t...
16      2020-04-03  BC's Medical Health Officer says the province ...
...            ...                                                ...
536149  2020-04-03  "These days of pain and sadness are bringing m...
536151  2020-04-03  Did #POTUS just lie that many states operate D...
536152  2020-04-03  Last day of @GarrisonMillES Spirit Week. It ma...
536153  2020-04-03  This is a real concern for our USian community...
536156  2020-04-03  @nytimes Oh looky. now the #coronavirus is rac...

        favourites_count  retweet_count  followers_count
friends_count  \
8                  16531              0            42537
2761
10                  2975              4          7249747
125
13                   495              0            13438
510
14                  5450              6          4889749
776
16                   500              0             6934
2137
...                  ...            ...              ...             ..
.
536149                 4              0                0
29
536151             10978              1             5531
5255
536152               699              1              103
102
536153              9373              8            40533
6687
536156              4633              0              105
212

       account_created_at  verified  follower_classification  \
8              2009-05-01         0                        1
10             2009-04-29         1                        1
```

```
13             2011-01-27         1                          1
14             2009-06-04         1                          1
16             2010-07-27         0                          1
...                   ...       ...                        ...
536149         2020-04-03         0                          0
536151         2010-04-02         0                          1
536152         2013-07-16         0                          0
536153         2010-06-07         0                          1
536156         2019-06-20         0                          0

                                                 processed_text
8       [constant, shift, natur, public, health, crisi...
10      [iran, parliament, speaker, test, posit, https...
13            [domest, violenc, start, dure, stop, https]
14      [show, must, go, lloyd, webber, music, air, fr...
16      [bc, medic, health, offic, say, provinc, conti...
...                                                   ...
536149  [day, pain, sad, bring, mani, hidden, problem,...
536151  [potus, lie, mani, state, oper, dos, base, sys...
536152  [last, day, garrisonmill, spirit, week, may, v...
536153  [real, concern, usian, communiti, member, avoi...
536156     [nytim, oh, looki, coronavirus, racist, https]

[266926 rows x 10 columns]
```

1.3.2.2: English Sentiment Analysis Calculation Using VaderSentiment

For our sentiment analysis calculation, we will be using the VaderSentiment library. (https://github.com/cjhutto/vaderSentiment)

VaderSentiment is a sentiment analysis and natural language processing toolkit trained on tweets that has been extensively used to investigate sentiment in a wide variety of situations and in many academic studies.

For example, it has been used to investigate tweet sentiment in Canada (https://www.ssph-journal.org/articles/10.3389/ijph.2022.1605241/full), attitudes towards Bitcoin on Twitter (https://www.mdpi.com/2504-2289/4/4/33), health-related social media data (https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8061710/), and more.

Thus, we felt that for the purposes of our analysis, which is to investigate multilingual sentiment in tweets regarding COVID-19, that the VaderSentiment toolkit would be a good fit. Because of its usage by multiple studies in reputable journals, and its origin as a model trained on tweet data fitting our current analysis, this toolkit was a good fit for what we wished to do.

```
# Installs the vaderSentiment toolkit
!pip install vaderSentiment
```

```
Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: vaderSentiment in
```

```
/usr/local/lib/python3.8/dist-packages (3.3.2)
Requirement already satisfied: requests in
/usr/local/lib/python3.8/dist-packages (from vaderSentiment) (2.23.0)
Requirement already satisfied: idna<3,>=2.5 in
/usr/local/lib/python3.8/dist-packages (from requests->vaderSentiment)
(2.10)
Requirement already satisfied: chardet<4,>=3.0.2 in
/usr/local/lib/python3.8/dist-packages (from requests->vaderSentiment)
(3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1
in /usr/local/lib/python3.8/dist-packages (from requests-
>vaderSentiment) (1.24.3)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.8/dist-packages (from requests->vaderSentiment)
(2022.9.24)
```

```
# Downloads the vader lexicon from NLTK package
nltk.download('vader_lexicon')
```

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

```
True
```

```
# Import the SentimentIntensityAnalyzer class
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer

# Create a SentimentIntensityAnalyzer object
sia = SentimentIntensityAnalyzer()

# Create a user-defined function that finds the compound sentiment
score for the entire sentence
def retrieve_sentiment(content):
  sentence = ' '.join(word for word in content)
  return sia.polarity_scores(sentence)['compound']

# Run the function to find sentiment for each processed tweet and add
it to a new sentiment column in english_tweets_df
english_tweets_df['sentiment'] =
english_tweets_df['processed_text'].apply(lambda x:
retrieve_sentiment(x))

# Takes the head of the english_tweets_df to investigate how it
appears currently
english_tweets_df.head(250)
```

```
     created_at                                              text  \
8    2020-04-03  "The constantly shifting nature of a public he...
10   2020-04-03  Iran parliament speaker tests positive for #CO...
13   2020-04-03  “Domestic violence did not start during #COVID...
14   2020-04-03  The shows must go on!: Lloyd Webber musicals t...
```

```
16   2020-04-03  BC's Medical Health Officer says the province ...
..          ...                                                ...
454  2020-04-03  Note  #COVID19 patients on #ICU are frequently...
455  2020-04-03  “It's time to reimagine everything. How do we ...
456  2020-04-03  "#COVID19 is the virus. Capitalism is the cris...
457  2020-04-03  We have shared new guidance with funeral indus...
458  2020-04-03  “As Americans, we always come together and pul...

     favourites_count  retweet_count  followers_count
friends_count  \
8               16531              0            42537            2761

10               2975              4          7249747             125

13                495              0            13438             510

14               5450              6          4889749             776

16                500              0             6934            2137

..                ...            ...              ...             ...

454             32651              0             7598             592

455              4219              0             1343             527

456              6533              0             1156            1395

457              9310              0             4851            2223

458                43              0              268             132


    account_created_at  verified  follower_classification  \
8           2009-05-01         0                         1
10          2009-04-29         1                         1
13          2011-01-27         1                         1
14          2009-06-04         1                         1
16          2010-07-27         0                         1
..                 ...       ...                       ...
454         2012-05-27         0                         1
455         2016-08-01         0                         1
456         2010-04-21         0                         1
457         2011-02-22         1                         1
458         2013-10-21         0                         0

                                      processed_text  sentiment
8    [constant, shift, natur, public, health, crisi...     0.4404
10   [iran, parliament, speaker, test, posit, https...     0.0000
13         [domest, violenc, start, dure, stop, https]    -0.2960
14   [show, must, go, lloyd, webber, music, air, fr...     0.5106
```

```
16   [bc, medic, health, offic, say, provinc, conti...     0.0000
..                                                 ...        ...
454  [note, patient, icu, frequent, run, dri, aggre...     0.0516
455  [time, reimagin, everyth, love, communiti, rei...     0.7650
456  [virus, capit, crisi, organ, solut, loan, tran...     0.0000
457  [share, new, guidanc, funer, industri, profess...     0.2960
458  [american, alway, come, togeth, pull, crisi, e...    -0.1027

[250 rows x 11 columns]
```

```
# We look at the dtypes of each column in order to split into
numerical and categorical
english_tweets_df.dtypes
```

```
created_at                  object
text                        object
favourites_count             int64
retweet_count                int64
followers_count              int64
friends_count                int64
account_created_at          object
verified                     int64
follower_classification      int64
processed_text              object
sentiment                  float64
dtype: object
```

```
# We take the numerical data, excluding followers_count, which
follower_classification accounts for, to investigate multicolinearity
english_numerics_df = english_tweets_df[['favourites_count',
'retweet_count', 'follower_classification',
                                         'friends_count', 'verified',
'sentiment']]
english_numerics_df
```

```
        favourites_count  retweet_count  follower_classification  \
8                  16531              0                        1
10                  2975              4                        1
13                   495              0                        1
14                  5450              6                        1
16                   500              0                        1
...                  ...            ...                      ...
536149                 4              0                        0
536151             10978              1                        1
536152               699              1                        0
536153              9373              8                        1
536156              4633              0                        0

        friends_count  verified  sentiment
8                2761         0     0.4404
```

```
10                 125          1     0.0000
13                 510          1    -0.2960
14                 776          1     0.5106
16                2137          0     0.0000
...                ...        ...        ...
536149              29          0    -0.8442
536151            5255          0     0.0000
536152             102          0     0.6124
536153            6687          0     0.2500
536156             212          0    -0.6124

[266926 rows x 6 columns]
```

1.3.2.3 English Processing for Training Data

For the purpose of training, we want to clean the Tweet such that it becomes one string of tokenize words, without miscellaneous characters such as '#' or '@' which may interfere with the model.

```
# User-defined function to clean the English tweets to get a tweet
ready for the model
@numba.jit
def clean_tweet_english(content):
  final_string = ""
  tokens = nltk.word_tokenize(content)
  for word in tokens:
    stemmed = word.lower()
    if(stemmed not in stopwords):
      for char in range(0, len(stemmed)):
        stemmed = stemmed.replace('#', '')
        stemmed = stemmed.replace('@', '')
      final_string = final_string + stemmed + " "
    else:
      continue
  return final_string

english_tweets_df['training_text'] =
english_tweets_df['text'].apply(lambda x: clean_tweet_english(x))
```

```
<ipython-input-73-43e2f4b252d0>:2: NumbaWarning:
Compilation is falling back to object mode WITH looplifting enabled
because Function "clean_tweet_english" failed type inference due to:
Unknown attribute 'word_tokenize' of type Module(<module 'nltk' from
'/usr/local/lib/python3.8/dist-packages/nltk/__init__.py'>)

File "<ipython-input-73-43e2f4b252d0>", line 5:
def clean_tweet_english(content):
    <source elided>
  final_string = ""
  tokens = nltk.word_tokenize(content)
```

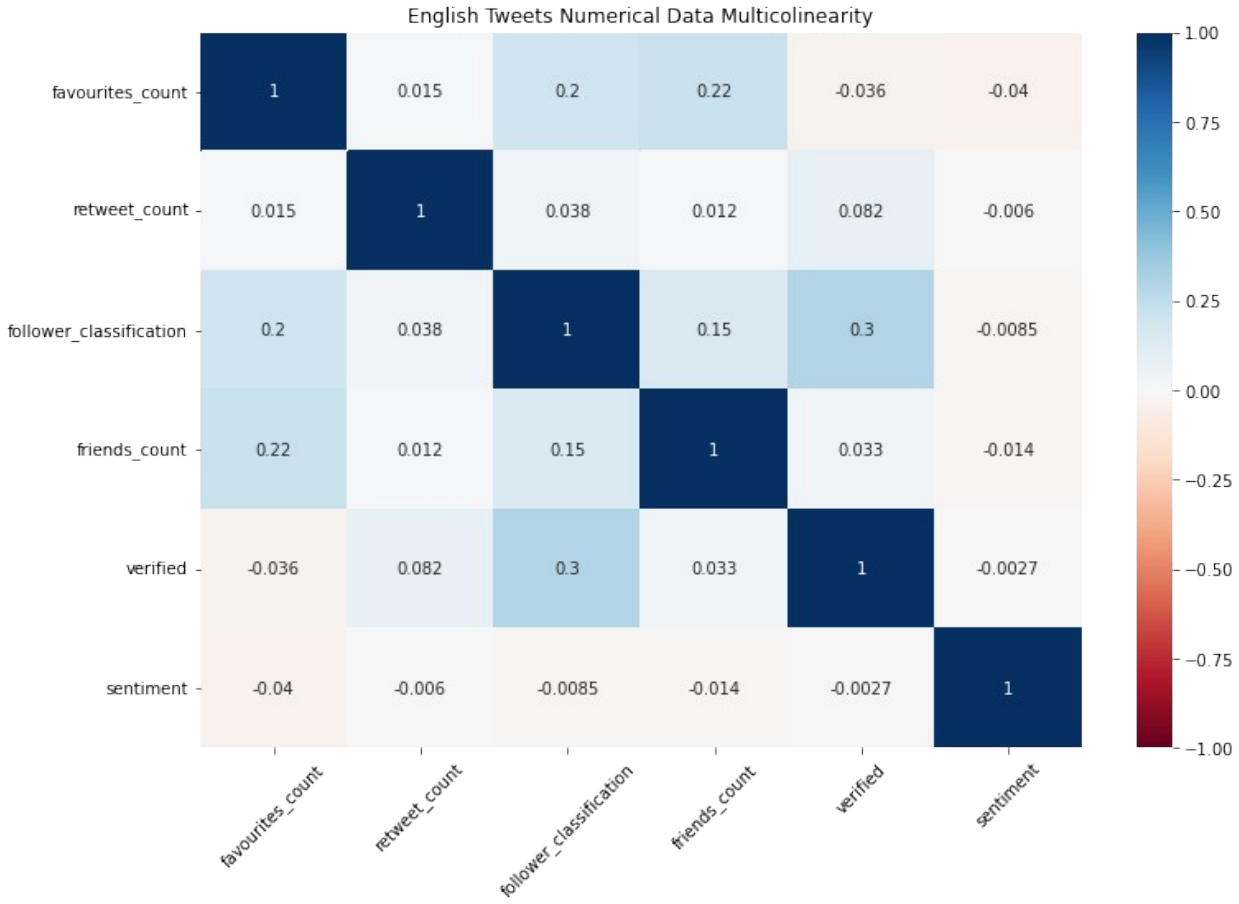
```
  ^

During: typing of get attribute at <ipython-input-73-43e2f4b252d0> (5)

File "<ipython-input-73-43e2f4b252d0>", line 5:
def clean_tweet_english(content):
    <source elided>
  final_string = ""
  tokens = nltk.word_tokenize(content)
  ^

  @numba.jit
<ipython-input-73-43e2f4b252d0>:2: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "clean_tweet_english" failed type inference due to:
Cannot determine Numba type of <class
'numba.core.dispatcher.LiftedLoop'>

File "<ipython-input-73-43e2f4b252d0>", line 6:
def clean_tweet_english(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "clean_tweet_english" was compiled in
object mode without forceobj=True, but has lifted loops.

File "<ipython-input-73-43e2f4b252d0>", line 4:
def clean_tweet_english(content):
  final_string = ""
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-73-43e2f4b252d0>", line 4:
def clean_tweet_english(content):
  final_string = ""
  ^
```

```
  warnings.warn(errors.NumbaDeprecationWarning(msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "clean_tweet_english" was compiled in
object mode without forceobj=True.

File "<ipython-input-73-43e2f4b252d0>", line 6:
def clean_tweet_english(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-73-43e2f4b252d0>", line 6:
def clean_tweet_english(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,
```

1.3.3: English Tweets Visualizations

We want to visualize all of the data within the English tweets dataset, so that we can find trends in our data, investigate multicolinearity, overall sentiment, etc.

```
# We create a correlation matrix heatmap to investigate
multicolinearity
# Seeing the result, we see that none of the variables are overly
colinear, meaning we can investigate all of them for our analysis
plt.figure(figsize=(12,8))
english_corr_matrix = sns.heatmap(english_numerics_df.corr(), vmin=-1,
vmax=1, cmap='RdBu', annot=True)
plt.title('English Tweets Numerical Data Multicolinearity')
plt.xticks(rotation=45)

(array([0.5, 1.5, 2.5, 3.5, 4.5, 5.5]),
 <a list of 6 Text major ticklabel objects>)
```



As we see from the correlation matrix heatmap, there is not high colinearity between any of the variables, so none have to be removed for the modeling.

```
english_tweets_df['sentiment'].describe()
```

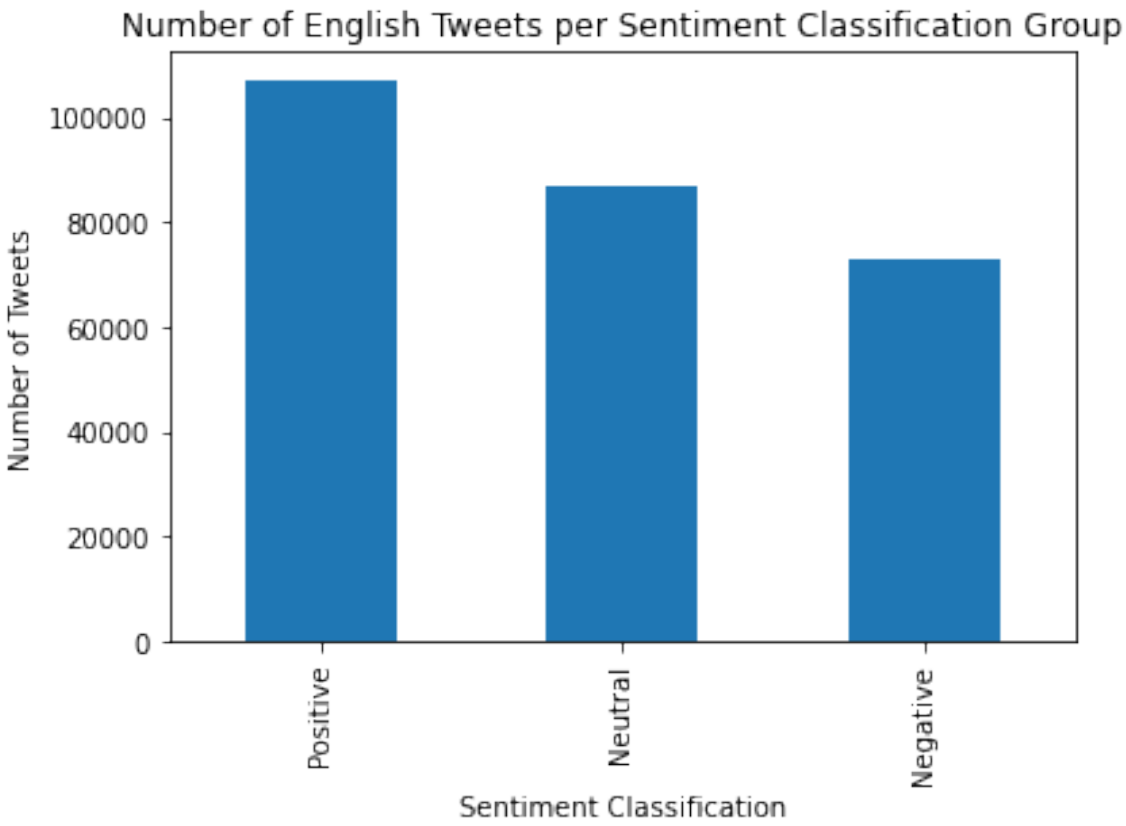
```
count    266926.000000
mean          0.073468
std           0.456280
min          -0.997000
25%          -0.177900
50%           0.000000
75%           0.421500
max           0.986700
Name: sentiment, dtype: float64
```

```
# Creates a new dataframe splitting the sentiment values into
Negative/Neutral/Positive based on sentiment compound value.
english_sentiment_df = english_tweets_df['sentiment'].apply(lambda x:
'Negative' if x <= -0.1 else 'Neutral' if -0.1 < x < 0.1 else
'Positive')
english_sentiment_df = english_sentiment_df.to_frame()
english_sentiment_df
```

```
       sentiment
8       Positive
10       Neutral
13      Negative
14      Positive
16       Neutral
...          ...
536149  Negative
536151   Neutral
536152  Positive
536153  Positive
536156  Negative

[266926 rows x 1 columns]
```

```
# Creates a bar plot visualizing the number of tweets per sentiment
classification group
english_sentiment_df['sentiment'].value_counts().plot(kind='bar')
plt.xlabel('Sentiment Classification')
plt.ylabel('Number of Tweets')
plt.title('Number of English Tweets per Sentiment Classification
Group')
```

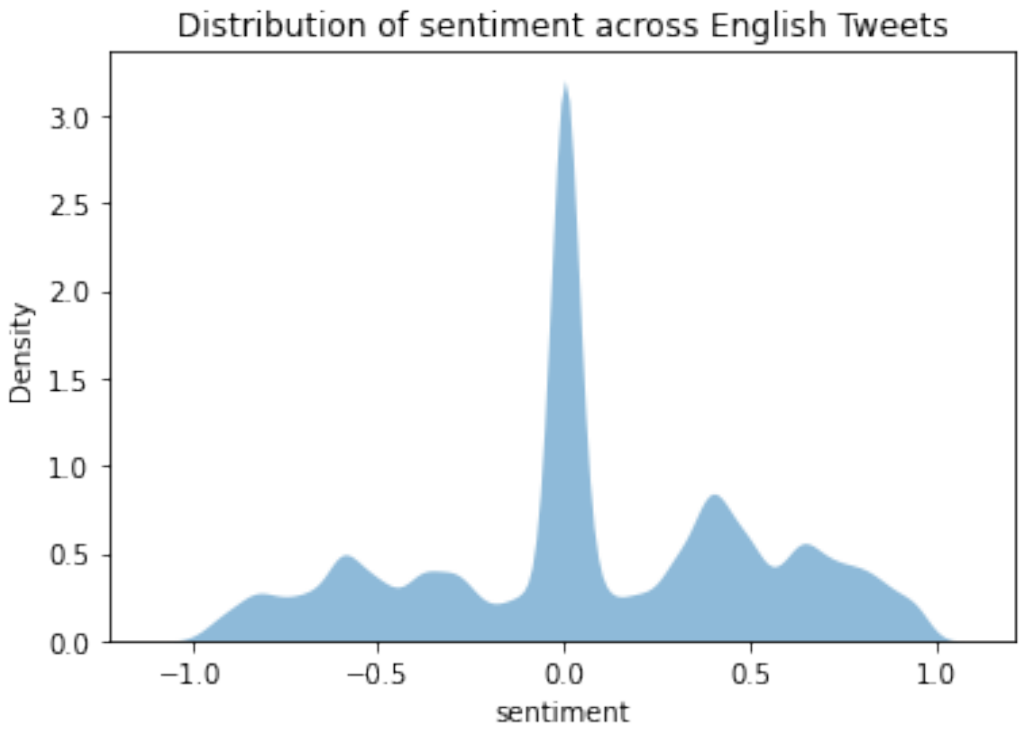
```
Text(0.5, 1.0, 'Number of English Tweets per Sentiment Classification
Group')
```



In the following plots, we utilize Kernel Density Estimation plots to visualize the distribution of sentiment across different Tweet demographics. KDE plots use a Gaussian kernel to plot a smooth distribution.

```
#Creates a KDE plot of sentiment across all English tweets
sns.kdeplot(data = english_numerics_df, x='sentiment',
            fill=True, alpha=0.5, linewidth=0
            ).set(title = 'Distribution of sentiment across English
Tweets')
```

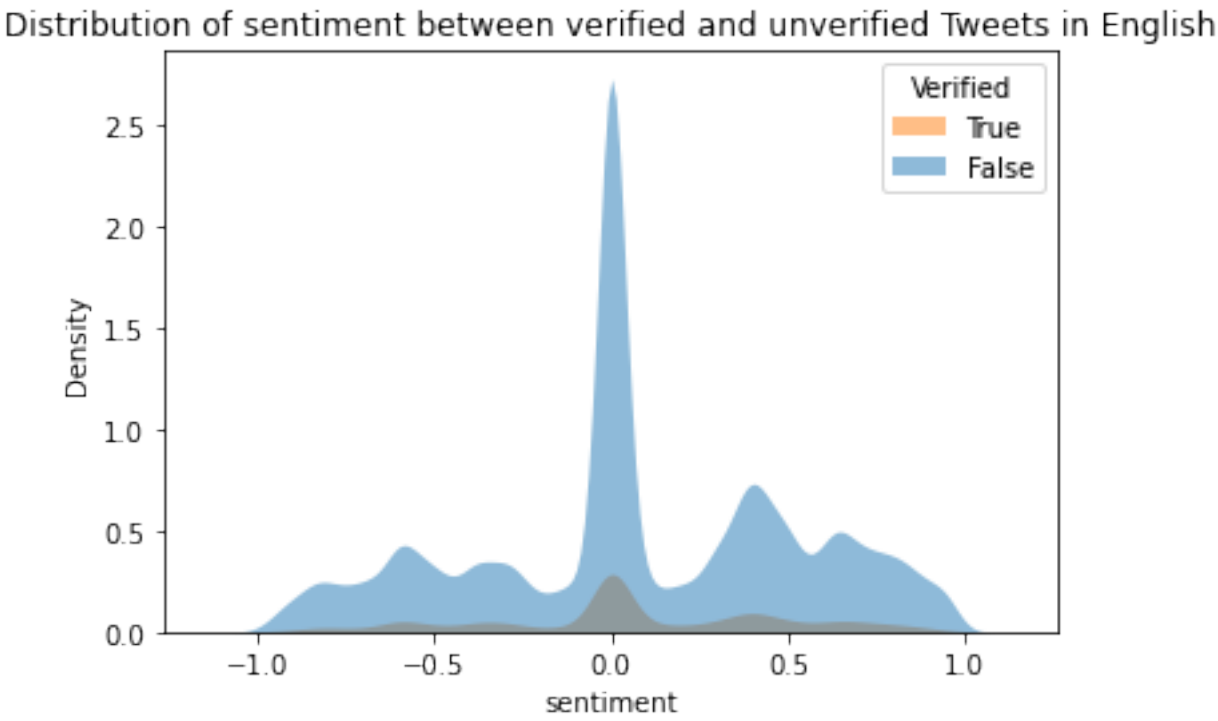
```
[Text(0.5, 1.0, 'Distribution of sentiment across English Tweets')]
```



From this KDE plot we can see that majority of tweets are neutral, with another significant spike in the 0.4 range of sentiment. Overall, there are more positive tweets with few tweets on either extreme.

```
#Compares sentiment between verified and unverified English tweets
sns.kdeplot(data = english_numerics_df, x='sentiment', hue='verified',

            fill=True, alpha=0.5, linewidth=0
            ).set(title='Distribution of sentiment between verified
and unverified Tweets in English')
plt.legend(title='Verified', labels=['True', 'False'])
```

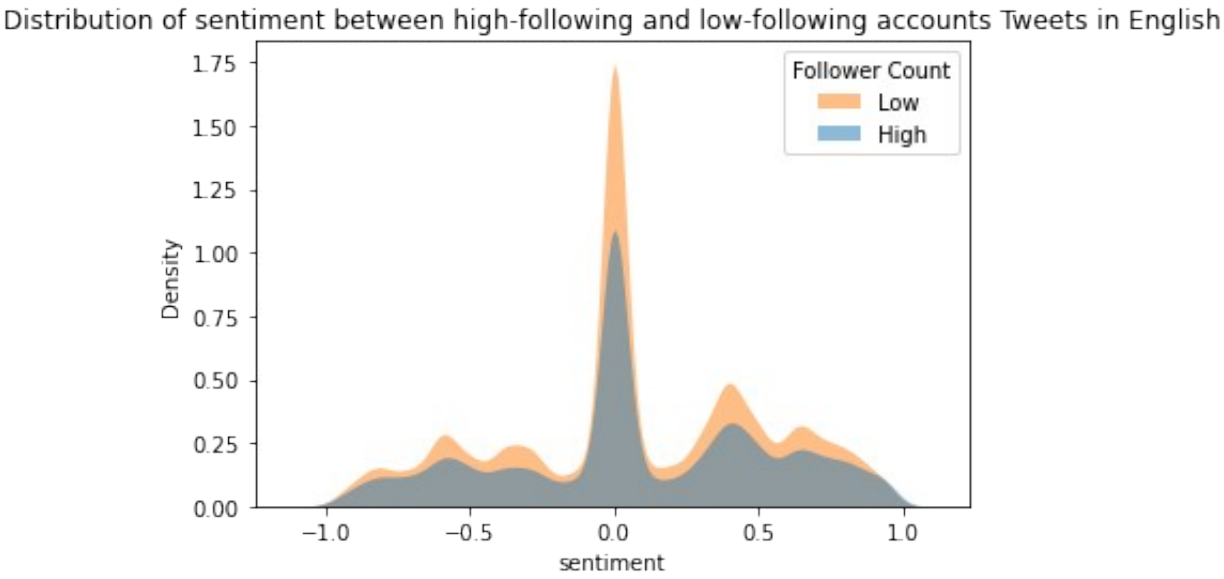
```
<matplotlib.legend.Legend at 0x7f582ef68bb0>
```



The verified distribution largely mirrors the unverified distribution, with the exception that more verified tweets compose of the 0.4 sentiment area than expected. Verified tweets are largely positive, with very few negative sentiment tweets.

```
sns.kdeplot(data = english_numerics_df, x='sentiment',
hue='follower_classification',
            fill=True, alpha=0.5, linewidth=0
            ).set(title='Distribution of sentiment between high-
following and low-following accounts Tweets in English')
plt.legend(title='Follower Count', labels=['Low', 'High'])
```

```
<matplotlib.legend.Legend at 0x7f582f693520>
```



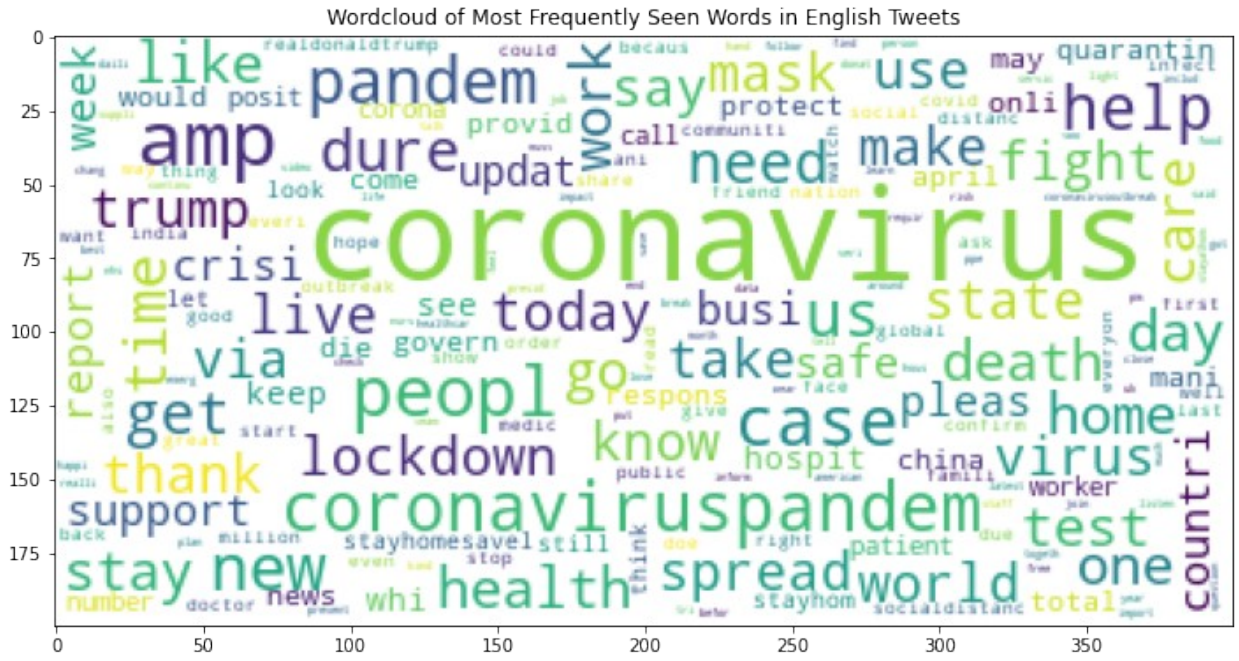
High and low following accounts have approximately the same distribution, which suggests there are no trends between these two demographics.

Next we will create a word cloud of the most common tokens. This can help us get a bigger picture of common trends amongst tweets, and allow us to guess the overall sentiment.

```
# Create the top tokens from our tokenized text. Due to the way our
tokenizer processed French,
# "https" is treated as a token and became the most popular token. To
offset this, we only count
# tokens if they're not equal to "https"
top_tokens_list_english = english_tweets_df['processed_text']
top_tokens_english = []
for sublist in top_tokens_list_english:
    for element in sublist:
      if(element != 'https'):
        top_tokens_english.append(element)

#Count each token and collect the top 20 most frequent ones
from collections import Counter
cnt = Counter()
for word in top_tokens_english:
    cnt[word] += 1
top_most_common_english = cnt.most_common(20)

#Plot the word cloud
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
w = WordCloud(background_color='white')

cnt2 = Counter()
for word in top_tokens_english:
    cnt2[word] += 1
```

```
plt.figure(figsize=(12, 8))
w.generate_from_frequencies(cnt2)
plt.title('Wordcloud of Most Frequently Seen Words in English Tweets')
plt.imshow(w)
```

```
<matplotlib.image.AxesImage at 0x7f582f5f8820>
```



This word cloud of the most frequent tokens suggests somewhat negative overall sentiment, with words such as "death", "lockdown", and "trump" hovering around the mid-size category. However, the largest words such as "coronavirus", "people", and "support" suggest a neutral sentiment.

1.4: Spanish Tweets Analysis

1.4.1: Spanish Tweets Pre-processing and EDA

In a similar fashion to how we investigated and removed unnecessary columns and nulls from the English tweet dataframe, we now want to do the same to the Spanish tweets using the dataframe we created earlier, spanish_tweets_df.

The steps used to process the Spanish tweets dataframe are mostly similar to that of the English tweets, albeit with differences, mainly in the sentiment analysis area.

```
# Prints the head of 20 rows to see how the Spanish tweets dataframe
looks like
spanish_tweets_df.head(20)
```

```
               status_id              user_id           created_at  \
0   1245863586586439680            133184048  2020-04-03T00:00:00Z
1   1245863584417992704           2722502906  2020-04-03T00:00:00Z
2   1245863584799678464   775704843994353665  2020-04-03T00:00:00Z
3   1245863587307868160   860252856829587457  2020-04-03T00:00:00Z
4   1245863586892779523             88957440  2020-04-03T00:00:00Z
5   1245863586557079553            357050985  2020-04-03T00:00:00Z
7   1245863584644513796            299693451  2020-04-03T00:00:00Z
9   1245863584455725063            499044422  2020-04-03T00:00:00Z
11  1245863586947186688  1042498579909107712  2020-04-03T00:00:00Z
12  1245863586129461250            122114793  2020-04-03T00:00:00Z
18  1245863584686465024            757469910  2020-04-03T00:00:00Z
19  1245863587660181506            182181105  2020-04-03T00:00:00Z
21  1245863584359424000             81852072  2020-04-03T00:00:00Z
23  1245863585630355458           2279419382  2020-04-03T00:00:00Z
25  1245863584514502661             58606864  2020-04-03T00:00:00Z
27  1245863586221744128             38277042  2020-04-03T00:00:00Z
28  1245863585189937152            132225222  2020-04-03T00:00:00Z
30  1245863586963963904  1184969362660196352  2020-04-03T00:00:00Z
31  1245863587547144192           2370663266  2020-04-03T00:00:00Z
38  1245863586657796097            148414749  2020-04-03T00:00:00Z

        screen_name                                                text
\
0         EcuadorTV  #QuédateEnCasa | Mira estas creaciones
origina...
1      GradaNorteMX  Contra el #Coronavirus 😷🧼⭕\n\n🏥
#QuédateEnCas...
2    AutoSupplyNews  @HondaMexico extiende suspensión de sus planta...

3      IMSS_SanLuis  Con manos limpias, seguro estarás mejor. #Prev...

4         Imagen_Mx  🚨😳 Baja California suma cuatro muertos por
#CO...
5     diario24horas  Ante la emergencia por #COVID19, la Cofepris, ...

7       tvnnoticias  ¿Tienes preguntas sobre el #COVID19? \nEnvía t...

9   QuintanaRooHoy1  #COVID19 ☣\nRegistró el #Vaticano un nuevo ca...

11  ContraReplicaMX  Senadores plantearon medidas para atender las ...

12         EfektoTv  .@NapoleonGomezUr propuso que el @senadomexica...

18  pichinchauniver  🗞#Nacionales | La Junta de Beneficencia de
#Gu...
19      tcsnoticias  Médicos de Estados Unidos piden protección ant...

21       TabascoHOY  #COVID19 ☣\nCiudadanos de #Wuhan, cuna del br...

23        cambioweb  #02Abr | #Internacionales | Perú restringe aún...
```

```
25       noroestemx  Aunque la demanda de servicios de transporte p...
27      889Noticias  De acuerdo con @HLGatell, subsecretario de Sal...
28        SSalud_mx  #ConferenciaDePrensa sobre el #Coronavirus #CO...
30   MaiteeeRosales  @CuitlahuacGJ es reprobado por el pueblo Verac...
31   PoliciaCtagena  #PrevenciónYAcción De noche y de día, seguimos...
38   primeraplanamx  Los Patriotas de Nueva Inglaterra prestan su a...

                   source  reply_to_status_id  reply_to_user_id  \
0              TweetDeck                 NaN               NaN
1              TweetDeck                 NaN               NaN
2              TweetDeck                 NaN      2.019508e+08
3              TweetDeck                 NaN               NaN
4              TweetDeck                 NaN               NaN
5              TweetDeck                 NaN               NaN
7   Twitter Media Studio                 NaN               NaN
9              TweetDeck                 NaN               NaN
11             TweetDeck                 NaN               NaN
12             TweetDeck                 NaN               NaN
18             TweetDeck                 NaN               NaN
19             TweetDeck                 NaN               NaN
21             TweetDeck                 NaN               NaN
23             TweetDeck                 NaN               NaN
25             TweetDeck                 NaN               NaN
27             TweetDeck                 NaN               NaN
28             TweetDeck                 NaN               NaN
30   Twitter for Android        1.245864e+18      1.184969e+18
31  Twitter Media Studio                 NaN               NaN
38             TweetDeck                 NaN               NaN

   reply_to_screen_name  is_quote  ...  country_code  place_full_name
\
0                   NaN     False  ...           NaN              NaN
1                   NaN     False  ...           NaN              NaN
2           HondaMexico     False  ...           NaN              NaN
3                   NaN     False  ...           NaN              NaN
4                   NaN     False  ...           NaN              NaN
5                   NaN     False  ...           NaN              NaN
7                   NaN     False  ...           NaN              NaN
```

```
9                    NaN     False  ...           NaN                NaN
11                   NaN     False  ...           NaN                NaN
12                   NaN     False  ...           NaN                NaN
18                   NaN     False  ...           NaN                NaN
19                   NaN     False  ...           NaN                NaN
21                   NaN     False  ...           NaN                NaN
23                   NaN     False  ...           NaN                NaN
25                   NaN     False  ...           NaN                NaN
27                   NaN     False  ...           NaN                NaN
28                   NaN     False  ...           NaN                NaN
30       MaiteeeRosales     False  ...           NaN                NaN
31                   NaN     False  ...           NaN                NaN
38                   NaN     False  ...           NaN                NaN


    place_type followers_count friends_count account_lang  \
0          NaN          536720          1164          NaN
1          NaN            1847           252          NaN
2          NaN             538           780          NaN
3          NaN            1015            41          NaN
4          NaN          293891           269          NaN
5          NaN          357448           585          NaN
7          NaN          811421          1610          NaN
9          NaN           10074          2517          NaN
11         NaN           13367          2556          NaN
12         NaN           42850           645          NaN
18         NaN           51432           501          NaN
19         NaN          680213           897          NaN
21         NaN          264711          1202          NaN
23         NaN           26722           803          NaN
25         NaN          127676           446          NaN
27         NaN          263096           164          NaN
28         NaN          864291           216          NaN
30         NaN              48            42          NaN
31         NaN           11823           151          NaN
38         NaN           14977          4832          NaN

      account_created_at  verified  lang lang_count
```

```
0   2010-04-15T06:31:39Z      True    es      82558
1   2014-08-10T21:20:32Z     False    es      82558
2   2016-09-13T14:37:01Z     False    es      82558
3   2017-05-04T22:00:38Z     False    es      82558
4   2009-11-10T16:01:41Z      True    es      82558
5   2011-08-17T19:30:28Z      True    es      82558
7   2011-05-16T14:51:19Z      True    es      82558
9   2012-02-21T18:21:57Z     False    es      82558
11  2018-09-19T19:40:04Z     False    es      82558
12  2010-03-11T16:53:28Z     False    es      82558
18  2012-08-14T16:41:36Z     False    es      82558
19  2010-08-24T00:53:13Z      True    es      82558
21  2009-10-12T14:18:22Z     False    es      82558
23  2014-01-06T18:09:58Z     False    es      82558
25  2009-07-20T22:08:51Z     False    es      82558
27  2009-05-06T21:09:11Z      True    es      82558
28  2010-04-12T16:53:45Z      True    es      82558
30  2019-10-17T23:08:29Z     False    es      82558
31  2014-03-03T16:38:57Z      True    es      82558
38  2010-05-26T16:51:56Z     False    es      82558

[20 rows x 23 columns]
```

```
# Looking at the information per column, again like in tweets_df,
there are many missing values in the columns:
# reply_to_status_id, reply_to_user_id, reply_to_screen_name,
country_code, place_full_name, place_type, and account_lang
spanish_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 82558 entries, 0 to 536155
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   status_id             82558 non-null  int64
 1   user_id               82558 non-null  int64
 2   created_at            82558 non-null  object
 3   screen_name           82558 non-null  object
 4   text                  82558 non-null  object
 5   source                82558 non-null  object
 6   reply_to_status_id    7809 non-null   float64
 7   reply_to_user_id      9397 non-null   float64
 8   reply_to_screen_name  9397 non-null   object
 9   is_quote              82558 non-null  bool
 10  is_retweet            82558 non-null  bool
 11  favourites_count      82558 non-null  int64
 12  retweet_count         82558 non-null  int64
 13  country_code          3565 non-null   object
 14  place_full_name       3566 non-null   object
 15  place_type            3566 non-null   object
```

```
 16  followers_count       82558 non-null  int64
 17  friends_count         82558 non-null  int64
 18  account_lang          0 non-null      float64
 19  account_created_at    82558 non-null  object
 20  verified              82558 non-null  bool
 21  lang                  82558 non-null  object
 22  lang_count            82558 non-null  int64
dtypes: bool(3), float64(3), int64(7), object(10)
memory usage: 13.5+ MB
```

1.4.1.1: Drop Nulls in Spanish Tweets

Similarly to what we did with the English tweets, we want to investigate the columns with null values and address them. Looking at these, we drop the columns such as reply_to_user_id or place_full_name that have over 75% null values, just like in English tweets.

Again, these have an extremely large number of null values, for one, and cannot be easily imputated!

For example, if the country code or place is null, there's no easy imputation to be had. Likewise, whether or not it's a reply is rather meaningless in our analysis, so we can drop it rather than doing null imputation!

```
# Drops columns which have 75% or more of rows being null values
spanish_tweets_df.dropna(thresh=int(0.75 *
spanish_tweets_df.shape[0]), axis=1, inplace=True)
spanish_tweets_df
```

```
/usr/local/lib/python3.8/dist-packages/pandas/util/_decorators.py:311:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return func(*args, **kwargs)
```

```
                  status_id             user_id            created_at
\
0       1245863586586439680           133184048  2020-04-03T00:00:00Z

1       1245863584417992704          2722502906  2020-04-03T00:00:00Z

2       1245863584799678464  775704843994353665  2020-04-03T00:00:00Z

3       1245863587307868160  860252856829587457  2020-04-03T00:00:00Z

4       1245863586892779523            88957440  2020-04-03T00:00:00Z

...                     ...                 ...                   ...
```

```
536130  1246225952608325632           142811824  2020-04-03T23:59:55Z

536131  1246225954210537472            29655554  2020-04-03T23:59:55Z

536148  1246225966667628546           830542951  2020-04-03T23:59:58Z

536154  1246225970312462341          3820878372  2020-04-03T23:59:59Z

536155  1246225970379530245           374870492  2020-04-03T23:59:59Z


           screen_name
text  \
0            EcuadorTV  #QuédateEnCasa | Mira estas creaciones
origina...
1          GradaNorteMX  Contra el #Coronavirus 😷👏○□\n\n🏥
#QuédateEnCas...
2       AutoSupplyNews  @HondaMexico extiende suspensión de sus
planta...
3          IMSS_SanLuis  Con manos limpias, seguro estarás mejor.
#Prev...
4             Imagen_Mx  🤷‍♂️😷 Baja California suma cuatro muertos por
#CO...
...                  ...
...
536130  QuintaFuerzaMX  □😷 A partir del lunes circularán menos taxis
e...
536131        Servindi  #Brasil: ¿Puede #Bolsonaro ser destituido
por ...
536148       LaSandino  #UltimoMinuto Tercera muerte por #Covid19 es
c...
536154       dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
536155      cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
mundo,...

                    source  is_quote  is_retweet  favourites_count  \
0                TweetDeck     False       False              2307
1                TweetDeck     False       False              1473
2                TweetDeck     False       False               422
3                TweetDeck     False       False               349
4                TweetDeck     False       False              3640
...                    ...       ...         ...               ...
536130           TweetDeck     False       False              1322
536131      Twitter Web App     False       False              6753
536148  Twitter for Android     False       False              3886
536154  Twitter for Android      True       False                30
536155   Twitter for iPhone     False       False             26271

        retweet_count  followers_count  friends_count
```

```
account_created_at  \
0                  15           536720           1164  2010-04-
15T06:31:39Z
1                   0             1847            252  2014-08-
10T21:20:32Z
2                   0              538            780  2016-09-
13T14:37:01Z
3                   0             1015             41  2017-05-
04T22:00:38Z
4                   1           293891            269  2009-11-
10T16:01:41Z
...               ...              ...            ...
...
536130              0            19522            380  2010-05-
11T20:48:45Z
536131              3            21444           1937  2009-04-
08T06:22:05Z
536148              1            29675            126  2012-09-
18T07:23:40Z
536154              0               57             59  2015-10-
08T03:09:46Z
536155              0              349           1214  2011-09-
17T03:18:48Z

        verified lang  lang_count
0           True   es       82558
1          False   es       82558
2          False   es       82558
3          False   es       82558
4           True   es       82558
...          ...  ...         ...
536130     False   es       82558
536131     False   es       82558
536148     False   es       82558
536154     False   es       82558
536155     False   es       82558

[82558 rows x 16 columns]
```

1.4.1.2: Drop Duplicates in Spanish Tweets

We do not want duplicated tweets within our Spanish tweet dataset either. Similarly, if the same Spanish tweet is used multiple times, then its sentiment would be the same and that could skew and alter our overall sentiment that the model would be trained on for the Spanish tweets.

```
# Drops the duplicates in the Spanish tweets, keeping the first
element so we don't remove all of the tweets and their sentiment.
spanish_tweets_df.drop_duplicates(keep='first', inplace=True)
```

```
/usr/local/lib/python3.8/dist-packages/pandas/util/_decorators.py:311:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return func(*args, **kwargs)
```

```
# Look at resulting dataframe
spanish_tweets_df
```

```
                  status_id              user_id            created_at
\
0       1245863586586439680            133184048  2020-04-03T00:00:00Z

1       1245863584417992704           2722502906  2020-04-03T00:00:00Z

2       1245863584799678464  775704843994353665  2020-04-03T00:00:00Z

3       1245863587307868160  860252856829587457  2020-04-03T00:00:00Z

4       1245863586892779523             88957440  2020-04-03T00:00:00Z

...                     ...                  ...                   ...

536130  1246225952608325632            142811824  2020-04-03T23:59:55Z

536131  1246225954210537472             29655554  2020-04-03T23:59:55Z

536148  1246225966667628546            830542951  2020-04-03T23:59:58Z

536154  1246225970312462341           3820878372  2020-04-03T23:59:59Z

536155  1246225970379530245            374870492  2020-04-03T23:59:59Z


           screen_name
text  \
0            EcuadorTV  #QuédateEnCasa | Mira estas creaciones
origina...
1         GradaNorteMX  Contra el #Coronavirus 😷👏🏻\n\n🏠
#QuédateEnCas...
2       AutoSupplyNews  @HondaMexico extiende suspensión de sus
planta...
3         IMSS_SanLuis  Con manos limpias, seguro estarás mejor.
#Prev...
4            Imagen_Mx  🚨😷 Baja California suma cuatro muertos por
#CO...
...                ...
...
```

```
536130  QuintaFuerzaMX  🤔 A partir del lunes circularán menos taxis
e...
536131        Servindi  #Brasil: ¿Puede #Bolsonaro ser destituido
por ...
536148       LaSandino  #UltimoMinuto Tercera muerte por #Covid19 es
c...
536154       dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
536155      cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
mundo,...

                     source  is_quote  is_retweet  favourites_count  \
0                 TweetDeck     False       False              2307
1                 TweetDeck     False       False              1473
2                 TweetDeck     False       False               422
3                 TweetDeck     False       False               349
4                 TweetDeck     False       False              3640
...                     ...       ...         ...               ...
536130            TweetDeck     False       False              1322
536131      Twitter Web App     False       False              6753
536148  Twitter for Android     False       False              3886
536154  Twitter for Android      True       False                30
536155   Twitter for iPhone     False       False             26271

        retweet_count  followers_count  friends_count
account_created_at  \
0                  15           536720           1164  2010-04-
15T06:31:39Z
1                   0             1847            252  2014-08-
10T21:20:32Z
2                   0              538            780  2016-09-
13T14:37:01Z
3                   0             1015             41  2017-05-
04T22:00:38Z
4                   1           293891            269  2009-11-
10T16:01:41Z
...               ...              ...            ...
...
536130              0            19522            380  2010-05-
11T20:48:45Z
536131              3            21444           1937  2009-04-
08T06:22:05Z
536148              1            29675            126  2012-09-
18T07:23:40Z
536154              0               57             59  2015-10-
08T03:09:46Z
536155              0              349           1214  2011-09-
17T03:18:48Z

        verified lang  lang_count
```

```
0           True   es       82558
1          False   es       82558
2          False   es       82558
3          False   es       82558
4           True   es       82558
...          ...  ...         ...
536130     False   es       82558
536131     False   es       82558
536148     False   es       82558
536154     False   es       82558
536155     False   es       82558

[82550 rows x 16 columns]
```

1.4.1.3: Investigation of Spanish Source Column

Once again looking at the Spanish tweets dataset, we see that there are some null values within the column, the only column that applies to. Thus, we want to investigate the source column and decide how to deal with those rows, along with what to do with the source column in general.

```
# Finds a summary of spanish_tweets_df -- note the null values within
the source column.
spanish_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 82550 entries, 0 to 536155
Data columns (total 16 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   status_id           82550 non-null  int64
 1   user_id             82550 non-null  int64
 2   created_at          82550 non-null  object
 3   screen_name         82550 non-null  object
 4   text                82550 non-null  object
 5   source              82550 non-null  object
 6   is_quote            82550 non-null  bool
 7   is_retweet          82550 non-null  bool
 8   favourites_count    82550 non-null  int64
 9   retweet_count       82550 non-null  int64
 10  followers_count     82550 non-null  int64
 11  friends_count       82550 non-null  int64
 12  account_created_at  82550 non-null  object
 13  verified            82550 non-null  bool
 14  lang                82550 non-null  object
 15  lang_count          82550 non-null  int64
dtypes: bool(3), int64(7), object(6)
memory usage: 9.1+ MB
```

```
# Looks at the rows where the source column is null
spanish_tweets_df[spanish_tweets_df['source'].isna()]
```

```
Empty DataFrame
Columns: [status_id, user_id, created_at, screen_name, text, source,
is_quote, is_retweet, favourites_count, retweet_count,
followers_count, friends_count, account_created_at, verified, lang,
lang_count]
Index: []

# Looks at a few rows where the source column isn't null in comparison
spanish_tweets_df[~(spanish_tweets_df['source'].isna())]

                  status_id             user_id            created_at
\
0       1245863586586439680           133184048  2020-04-03T00:00:00Z

1       1245863584417992704          2722502906  2020-04-03T00:00:00Z

2       1245863584799678464  775704843994353665  2020-04-03T00:00:00Z

3       1245863587307868160  860252856829587457  2020-04-03T00:00:00Z

4       1245863586892779523            88957440  2020-04-03T00:00:00Z

...                     ...                 ...                   ...

536130  1246225952608325632           142811824  2020-04-03T23:59:55Z

536131  1246225954210537472            29655554  2020-04-03T23:59:55Z

536148  1246225966667628546           830542951  2020-04-03T23:59:58Z

536154  1246225970312462341          3820878372  2020-04-03T23:59:59Z

536155  1246225970379530245           374870492  2020-04-03T23:59:59Z


           screen_name
text  \
0            EcuadorTV  #QuédateEnCasa | Mira estas creaciones
origina...
1          GradaNorteMX  Contra el #Coronavirus 😷👏🏻□\n\n🏠
#QuédateEnCas...
2       AutoSupplyNews  @HondaMexico extiende suspensión de sus
planta...
3          IMSS_SanLuis  Con manos limpias, seguro estarás mejor.
#Prev...
4             Imagen_Mx  🚨😷 Baja California suma cuatro muertos por
#CO...
...                  ...
...
536130  QuintaFuerzaMX  □😷 A partir del lunes circularán menos taxis
e...
```

```
536131        Servindi  #Brasil: ¿Puede #Bolsonaro ser destituido
por ...
536148       LaSandino  #UltimoMinuto Tercera muerte por #Covid19 es
c...
536154       dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
536155      cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
mundo,...

                     source  is_quote  is_retweet  favourites_count  \
0                 TweetDeck     False       False              2307
1                 TweetDeck     False       False              1473
2                 TweetDeck     False       False               422
3                 TweetDeck     False       False               349
4                 TweetDeck     False       False              3640
...                     ...       ...         ...               ...
536130            TweetDeck     False       False              1322
536131      Twitter Web App     False       False              6753
536148  Twitter for Android     False       False              3886
536154  Twitter for Android      True       False                30
536155   Twitter for iPhone     False       False             26271

        retweet_count  followers_count  friends_count
account_created_at  \
0                  15           536720           1164  2010-04-
15T06:31:39Z
1                   0             1847            252  2014-08-
10T21:20:32Z
2                   0              538            780  2016-09-
13T14:37:01Z
3                   0             1015             41  2017-05-
04T22:00:38Z
4                   1           293891            269  2009-11-
10T16:01:41Z
...               ...              ...            ...
...
536130              0            19522            380  2010-05-
11T20:48:45Z
536131              3            21444           1937  2009-04-
08T06:22:05Z
536148              1            29675            126  2012-09-
18T07:23:40Z
536154              0               57             59  2015-10-
08T03:09:46Z
536155              0              349           1214  2011-09-
17T03:18:48Z

        verified lang  lang_count
0           True   es       82558
1          False   es       82558
```

```
2          False   es       82558
3          False   es       82558
4           True   es       82558
...          ...  ...         ...
536130     False   es       82558
536131     False   es       82558
536148     False   es       82558
536154     False   es       82558
536155     False   es       82558

[82550 rows x 16 columns]
```

```
# Investigates how many different sources for the tweets there are in
the Spanish ones.
spanish_tweets_df['source'].value_counts()
```

```
Twitter for Android    25500
Twitter Web App        22030
Twitter for iPhone     12912
TweetDeck               8321
Hootsuite Inc.          3002
                       ...
PortalPortuario            1
PlaguiBot                  1
Kontentino                 1
elperiodic.com             1
Territorio Bitcoin         1
Name: source, Length: 324, dtype: int64
```

```
# We see that there are many, making imputation not worth it, and is
too much data for proper trend analysis, so we drop the column.
spanish_tweets_df.drop(['source'], axis=1, inplace=True)
spanish_tweets_df
```

```
/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py:4906:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return super().drop(
```

```
                  status_id             user_id            created_at
\
0       1245863586586439680           133184048  2020-04-03T00:00:00Z

1       1245863584417992704          2722502906  2020-04-03T00:00:00Z

2       1245863584799678464  775704843994353665  2020-04-03T00:00:00Z
```

```
3       1245863587307868160  860252856829587457  2020-04-03T00:00:00Z

4       1245863586892779523            88957440  2020-04-03T00:00:00Z

...                     ...                 ...                   ...

536130  1246225952608325632          142811824  2020-04-03T23:59:55Z

536131  1246225954210537472           29655554  2020-04-03T23:59:55Z

536148  1246225966667628546          830542951  2020-04-03T23:59:58Z

536154  1246225970312462341         3820878372  2020-04-03T23:59:59Z

536155  1246225970379530245          374870492  2020-04-03T23:59:59Z


           screen_name
text  \
0            EcuadorTV  #QuédateEnCasa | Mira estas creaciones
origina...
1          GradaNorteMX  Contra el #Coronavirus 😷👏🏻□\n\n🏠
#QuédateEnCas...
2       AutoSupplyNews  @HondaMexico extiende suspensión de sus
planta...
3          IMSS_SanLuis  Con manos limpias, seguro estarás mejor.
#Prev...
4             Imagen_Mx  🚨😷 Baja California suma cuatro muertos por
#CO...
...                 ...
...
536130  QuintaFuerzaMX  □😷 A partir del lunes circularán menos taxis
e...
536131        Servindi  #Brasil: ¿Puede #Bolsonaro ser destituido
por ...
536148       LaSandino  #UltimoMinuto Tercera muerte por #Covid19 es
c...
536154       dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
536155      cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
mundo,...

        is_quote  is_retweet  favourites_count  retweet_count  \
0          False       False              2307             15
1          False       False              1473              0
2          False       False               422              0
3          False       False               349              0
4          False       False              3640              1
...          ...         ...               ...            ...
536130     False       False              1322              0
```

```
536131     False       False              6753              3
536148     False       False              3886              1
536154      True       False                30              0
536155     False       False             26271              0

        followers_count  friends_count    account_created_at  verified
lang  \
0                536720           1164  2010-04-15T06:31:39Z      True
es
1                  1847            252  2014-08-10T21:20:32Z     False
es
2                   538            780  2016-09-13T14:37:01Z     False
es
3                  1015             41  2017-05-04T22:00:38Z     False
es
4                293891            269  2009-11-10T16:01:41Z      True
es
...                 ...            ...                   ...       ...
...
536130           19522            380  2010-05-11T20:48:45Z     False
es
536131           21444           1937  2009-04-08T06:22:05Z     False
es
536148           29675            126  2012-09-18T07:23:40Z     False
es
536154              57             59  2015-10-08T03:09:46Z     False
es
536155             349           1214  2011-09-17T03:18:48Z     False
es

        lang_count
0            82558
1            82558
2            82558
3            82558
4            82558
...            ...
536130       82558
536131       82558
536148       82558
536154       82558
536155       82558

[82550 rows x 15 columns]
```

1.4.1.4: Investigation of Spanish Verified and Followers Count Columns

Just as in the English tweet dataset, we want to investigate the discrepancy between how the overall sentiment is for users who are verified versus users who are not verified in the Spanish tweet dataset.

Similarly, we again want to look at the difference between overall sentiment of users with high numbers of followers versus low numbers of followers. We will do cleaning and visualizations of the Spanish verified and followers_count columns to prepare for this avenue of investigation.

Looking at the data, it is interesting to note that the difference between the number of high-follower people and low-follower people is smaller than what the article indicates yet again, although this difference is larger than the one exhibited by the English tweets.

This difference could be a reflection of a bias present in the selection of the tweets for the dataset, as it makes sense that users who have high amounts of followers are those who post more and post more about relevant topics (like COVID-19). Thus, this could be an explanation for the trends seen in the comparison bar plots.

However, the difference being smaller than the one exhibited in the English dataset is interesting. It could possibly be just noise in the Kaggle dataset, or a reflection of more verified accounts talking about COVID-19 in Spanish overall.

```
# Creates a new dataframe of the number of Spanish verified vs non-
verified users
spanish_verified_df =
spanish_tweets_df['verified'].value_counts().reset_index()
spanish_verified_df
```

```
   index  verified
0  False     72106
1   True     10444
```

```
# Applies a function to the followers_count column, assigning 1 if
count > 500, representing high followers, and 0 otherwise (low
followers)
spanish_tweets_df['follower_classification'] =
spanish_tweets_df['followers_count'].apply(lambda x: 1 if x >= 500
else 0)
spanish_tweets_df
```

```
<ipython-input-96-12c4077b2c74>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  spanish_tweets_df['follower_classification'] =
spanish_tweets_df['followers_count'].apply(lambda x: 1 if x >= 500
else 0)
```

```
                  status_id             user_id            created_at
\
0       1245863586586439680           133184048  2020-04-03T00:00:00Z

1       1245863584417992704          2722502906  2020-04-03T00:00:00Z
```

```
2       1245863584799678464  775704843994353665  2020-04-03T00:00:00Z
3       1245863587307868160  860252856829587457  2020-04-03T00:00:00Z
4       1245863586892779523            88957440  2020-04-03T00:00:00Z
...                     ...                 ...                   ...
536130  1246225952608325632           142811824  2020-04-03T23:59:55Z
536131  1246225954210537472            29655554  2020-04-03T23:59:55Z
536148  1246225966667628546           830542951  2020-04-03T23:59:58Z
536154  1246225970312462341          3820878372  2020-04-03T23:59:59Z
536155  1246225970379530245           374870492  2020-04-03T23:59:59Z

            screen_name
text  \
0             EcuadorTV  #QuédateEnCasa | Mira estas 𝐜𝐫𝐞𝐚𝐜𝐢𝐨𝐧𝐞𝐬
𝐨𝐫𝐢𝐠𝐢𝐧𝐚...
1           GradaNorteMX  Contra el #Coronavirus 😷👏◯□\n\n🏠
#QuédateEnCas...
2         AutoSupplyNews  @HondaMexico extiende suspensión de sus
planta...
3           IMSS_SanLuis  Con manos limpias, seguro estarás mejor.
#Prev...
4             Imagen_Mx  🤯😳 Baja California suma cuatro muertos por
#CO...
...                  ...
...
536130  QuintaFuerzaMX  □😳 A partir del lunes circularán menos taxis
e...
536131        Servindi  #Brasil: ¿Puede #Bolsonaro ser destituido
por ...
536148       LaSandino  #UltimoMinuto Tercera muerte por #Covid19 es
c...
536154       dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
536155      cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
mundo,...

        is_quote  is_retweet  favourites_count  retweet_count  \
0          False       False              2307             15
1          False       False              1473              0
2          False       False               422              0
3          False       False               349              0
```

```
4          False       False              3640               1
...          ...         ...               ...             ...
536130     False       False              1322               0
536131     False       False              6753               3
536148     False       False              3886               1
536154      True       False                30               0
536155     False       False             26271               0

        followers_count  friends_count    account_created_at  verified
lang  \
0                536720           1164  2010-04-15T06:31:39Z      True
es
1                  1847            252  2014-08-10T21:20:32Z     False
es
2                   538            780  2016-09-13T14:37:01Z     False
es
3                  1015             41  2017-05-04T22:00:38Z     False
es
4                293891            269  2009-11-10T16:01:41Z      True
es
...                 ...            ...                   ...       ...
...
536130           19522            380  2010-05-11T20:48:45Z     False
es
536131           21444           1937  2009-04-08T06:22:05Z     False
es
536148           29675            126  2012-09-18T07:23:40Z     False
es
536154              57             59  2015-10-08T03:09:46Z     False
es
536155             349           1214  2011-09-17T03:18:48Z     False
es

        lang_count  follower_classification
0            82558                        1
1            82558                        1
2            82558                        1
3            82558                        1
4            82558                        1
...            ...                      ...
536130       82558                        1
536131       82558                        1
536148       82558                        1
536154       82558                        0
536155       82558                        0

[82550 rows x 16 columns]

# Inside of verified column, again sets value to 1 if verified and 0
otherwise
```

```
spanish_tweets_df['verified'] =
spanish_tweets_df['verified'].apply(lambda x: 0 if x == False else 1)
spanish_tweets_df
```

```
<ipython-input-97-780d7475e728>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  spanish_tweets_df['verified'] =
spanish_tweets_df['verified'].apply(lambda x: 0 if x == False else 1)
```

	status_id	user_id	created_at
0	1245863586586439680	133184048	2020-04-03T00:00:00Z
1	1245863584417992704	2722502906	2020-04-03T00:00:00Z
2	1245863584799678464	775704843994353665	2020-04-03T00:00:00Z
3	1245863587307868160	860252856829587457	2020-04-03T00:00:00Z
4	1245863586892779523	88957440	2020-04-03T00:00:00Z
...	...	...	...
536130	1246225952608325632	142811824	2020-04-03T23:59:55Z
536131	1246225954210537472	29655554	2020-04-03T23:59:55Z
536148	1246225966667628546	830542951	2020-04-03T23:59:58Z
536154	1246225970312462341	3820878372	2020-04-03T23:59:59Z
536155	1246225970379530245	374870492	2020-04-03T23:59:59Z

	screen_name	text
0	EcuadorTV	#QuédateEnCasa | Mira estas **creaciones origina**...
1	GradaNorteMX	Contra el #Coronavirus 😒👊◯□\n\n🏠 #QuédateEnCas...
2	AutoSupplyNews	@HondaMexico extiende suspensión de sus planta...
3	IMSS_SanLuis	Con manos limpias, seguro estarás mejor. #Prev...
4	Imagen_Mx	🚨😷 Baja California suma cuatro muertos por #CO...

```
...                ...
...
536130  QuintaFuerzaMX  🤔 A partir del lunes circularán menos taxis
e...
536131        Servindi  #Brasil: ¿Puede #Bolsonaro ser destituido
por ...
536148       LaSandino  #UltimoMinuto Tercera muerte por #Covid19 es
c...
536154       dushe_chi  #Covid_19\nPermanezcamos en casa sin caer en
l...
536155      cemgiraldo  @PaisMenosPeor Ni zombies en el fin del
mundo,...

        is_quote  is_retweet  favourites_count  retweet_count  \
0          False       False              2307             15
1          False       False              1473              0
2          False       False               422              0
3          False       False               349              0
4          False       False              3640              1
...          ...         ...               ...            ...
536130     False       False              1322              0
536131     False       False              6753              3
536148     False       False              3886              1
536154      True       False                30              0
536155     False       False             26271              0

        followers_count  friends_count    account_created_at  verified
lang  \
0                536720           1164  2010-04-15T06:31:39Z         1
es
1                  1847            252  2014-08-10T21:20:32Z         0
es
2                   538            780  2016-09-13T14:37:01Z         0
es
3                  1015             41  2017-05-04T22:00:38Z         0
es
4                293891            269  2009-11-10T16:01:41Z         1
es
...                 ...            ...                   ...       ...
...
536130           19522            380  2010-05-11T20:48:45Z         0
es
536131           21444           1937  2009-04-08T06:22:05Z         0
es
536148           29675            126  2012-09-18T07:23:40Z         0
es
536154              57             59  2015-10-08T03:09:46Z         0
es
536155             349           1214  2011-09-17T03:18:48Z         0
```

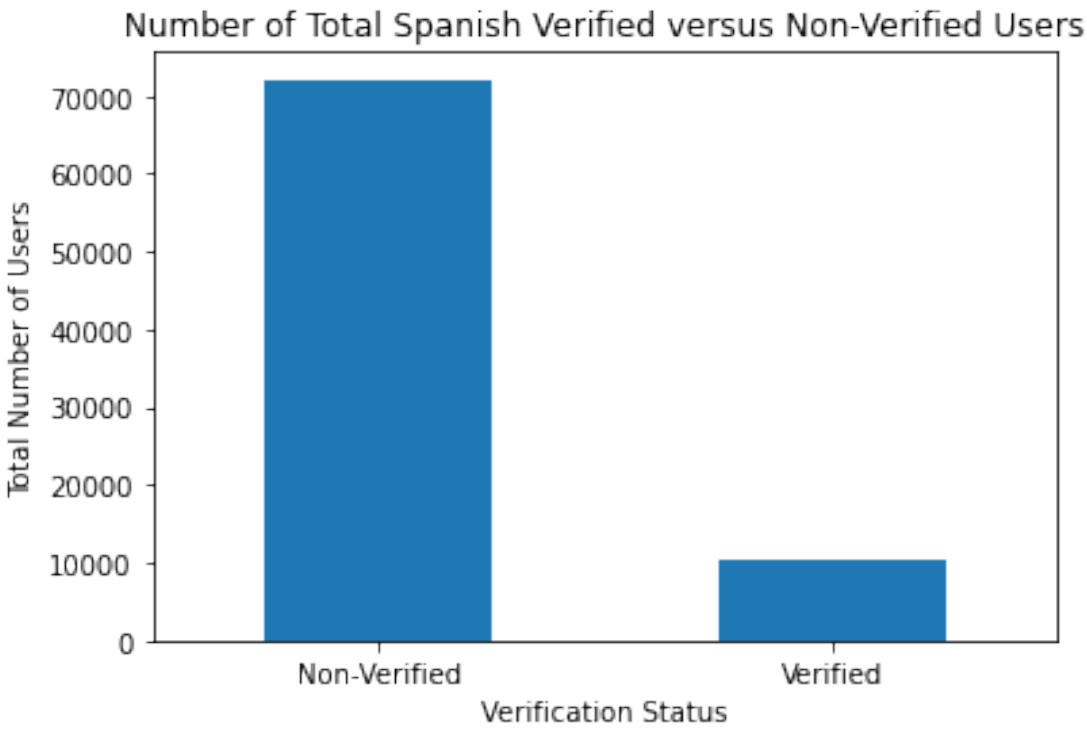
```
es
```

```
        lang_count  follower_classification
0            82558                        1
1            82558                        1
2            82558                        1
3            82558                        1
4            82558                        1
...            ...                      ...
536130       82558                        1
536131       82558                        1
536148       82558                        1
536154       82558                        0
536155       82558                        0

[82550 rows x 16 columns]
```

```
# Creates a new dataframe of the number of high follower vs low
follower users
spanish_follower_df =
spanish_tweets_df['follower_classification'].value_counts().reset_inde
x()
spanish_follower_df
```

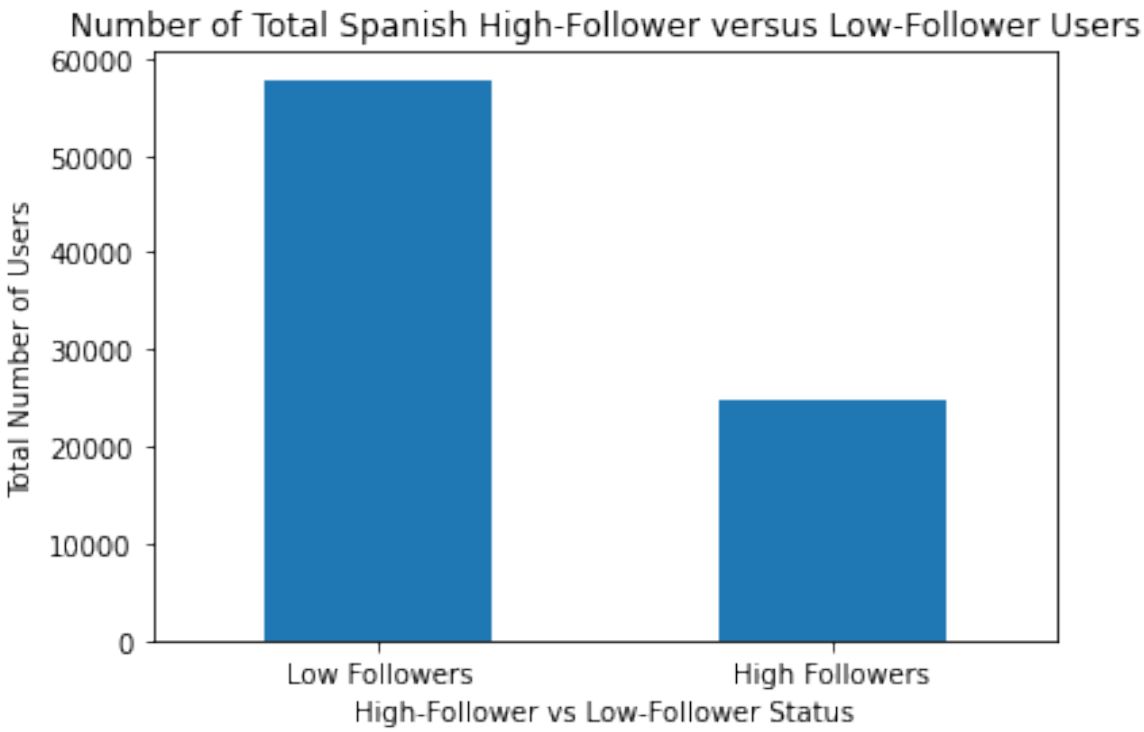
```
   index  follower_classification
0      1                     57804
1      0                     24746
```

```
# Plots number of Spanish verified vs non-verified tweets
spanish_verified_df.plot.bar(x='index', legend=None)
plt.title("Number of Total Spanish Verified versus Non-Verified
Users")
plt.ylabel("Total Number of Users")
plt.xlabel("Verification Status")

bars = ('Non-Verified', 'Verified')
x_pos = np.arange(len(bars))

plt.xticks(x_pos, bars)
plt.xticks(rotation=0)
plt.show()
```



This bar graph shows the large difference between the number of non-verified and verified users -- this makes sense, as naturally there would be less people that are verified, and both classifications still have large sample sizes to try and run analysis on. Even the verified status still has over 10,000 tweets to run analysis on.

```
# Plots number of Spanish high-follower vs low-follower tweets
spanish_follower_df.plot.bar(x='index', legend=None)
plt.title("Number of Total Spanish High-Follower versus Low-Follower
Users")
plt.ylabel("Total Number of Users")
plt.xlabel("High-Follower vs Low-Follower Status")

bars = ('Low Followers', 'High Followers')
x_pos = np.arange(len(bars))

plt.xticks(x_pos, bars)
plt.xticks(rotation=0)
plt.show()
```



This bar graph shows the difference between the number of high-follower users and low-follower users for the Spanish tweets. This is actually surprising, as given the article it indicated that "high-follower" users were only ~1-2% of the total Twitter userbase. However, the mass majority of Twitter accounts most likely do not post at all, so if a working usable Kaggle dataset was to be compiled, then it is understandable to have an overrepresentation of accounts that post often, and therefore high-follower accounts by association. Also, this matches more with the predicitons given by the article, as there's a higher difference between the number of low followers and high followers. This could perhaps reflect a difference in how Spanish-speaking twitter addressed COVID-19 through their tweets.

1.4.1.5: Cleaning Spanish Tweets Time Stamps

Looking at the Spanish created_at and account_created_at columns, we notice that they need to be converted to proper datetime formatting, as right now they're in an improper format. Thus, we again need to reformat them and transform them to datetime format.

```
# Cleans the created_at column of spanish_tweets_df and converts it to
datetime format
spanish_tweets_df['created_at'] =
spanish_tweets_df['created_at'].apply(lambda x: x.split('T')[0])
pd.to_datetime(spanish_tweets_df['created_at'])
spanish_tweets_df
```

```
<ipython-input-101-df0e42ed52c3>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  spanish_tweets_df['created_at'] =
spanish_tweets_df['created_at'].apply(lambda x: x.split('T')[0])

                   status_id             user_id  created_at
screen_name  \
0       1245863586586439680           133184048  2020-04-03
EcuadorTV
1       1245863584417992704          2722502906  2020-04-03
GradaNorteMX
2       1245863584799678464  775704843994353665  2020-04-03
AutoSupplyNews
3       1245863587307868160  860252856829587457  2020-04-03
IMSS_SanLuis
4       1245863586892779523            88957440  2020-04-03
Imagen_Mx
...                      ...                 ...         ...
...
536130  1246225952608325632           142811824  2020-04-03
QuintaFuerzaMX
536131  1246225954210537472            29655554  2020-04-03
Servindi
536148  1246225966667628546           830542951  2020-04-03
LaSandino
536154  1246225970312462341          3820878372  2020-04-03
dushe_chi
536155  1246225970379530245           374870492  2020-04-03
cemgiraldo

                                                     text  is_quote  \
0       #QuédateEnCasa | Mira estas creaciones origina...     False

1       Contra el #Coronavirus 😒🧼◯□\n\n🏠 #QuédateEnCas...     False

2       @HondaMexico extiende suspensión de sus planta...     False
3       Con manos limpias, seguro estarás mejor. #Prev...     False
4       🚨😳 Baja California suma cuatro muertos por #CO...     False

...                                                   ...       ...
536130  □😳 A partir del lunes circularán menos taxis e...     False

536131  #Brasil: ¿Puede #Bolsonaro ser destituido por ...     False
536148  #UltimoMinuto Tercera muerte por #Covid19 es c...     False
536154  #Covid_19\nPermanezcamos en casa sin caer en l...      True
536155  @PaisMenosPeor Ni zombies en el fin del mundo,...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
```

```
0             False               2307              15           536720
1             False               1473               0             1847
2             False                422               0              538
3             False                349               0             1015
4             False               3640               1           293891
...             ...                ...             ...              ...
536130        False               1322               0            19522
536131        False               6753               3            21444
536148        False               3886               1            29675
536154        False                 30               0               57
536155        False              26271               0              349

        friends_count    account_created_at  verified lang  lang_count
\
0                1164  2010-04-15T06:31:39Z         1   es       82558
1                 252  2014-08-10T21:20:32Z         0   es       82558
2                 780  2016-09-13T14:37:01Z         0   es       82558
3                  41  2017-05-04T22:00:38Z         0   es       82558
4                 269  2009-11-10T16:01:41Z         1   es       82558
...               ...                   ...       ...  ...         ...
536130            380  2010-05-11T20:48:45Z         0   es       82558
536131           1937  2009-04-08T06:22:05Z         0   es       82558
536148            126  2012-09-18T07:23:40Z         0   es       82558
536154             59  2015-10-08T03:09:46Z         0   es       82558
536155           1214  2011-09-17T03:18:48Z         0   es       82558

        follower_classification
0                             1
1                             1
```

```
2                               1
3                               1
4                               1
...                           ...
536130                          1
536131                          1
536148                          1
536154                          0
536155                          0

[82550 rows x 16 columns]
```

```
# Cleans the account_created_at column of spanish_tweets_df and
converts it to datetime format
spanish_tweets_df['account_created_at'] =
spanish_tweets_df['account_created_at'].apply(lambda x: x.split('T')
[0])
pd.to_datetime(spanish_tweets_df['account_created_at'])
spanish_tweets_df
```

```
<ipython-input-102-5847b673f7bb>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  spanish_tweets_df['account_created_at'] =
spanish_tweets_df['account_created_at'].apply(lambda x: x.split('T')
[0])
```

```
                   status_id             user_id  created_at
screen_name  \
0       1245863586586439680           133184048  2020-04-03
EcuadorTV
1       1245863584417992704          2722502906  2020-04-03
GradaNorteMX
2       1245863584799678464  775704843994353665  2020-04-03
AutoSupplyNews
3       1245863587307868160  860252856829587457  2020-04-03
IMSS_SanLuis
4       1245863586892779523            88957440  2020-04-03
Imagen_Mx
...                     ...                 ...         ...
...
536130  1246225952608325632           142811824  2020-04-03
QuintaFuerzaMX
536131  1246225954210537472            29655554  2020-04-03
Servindi
536148  1246225966667628546           830542951  2020-04-03
```

```
LaSandino
536154  1246225970312462341          3820878372  2020-04-03
dushe_chi
536155  1246225970379530245           374870492  2020-04-03
cemgiraldo

                                                      text  is_quote  \
0       #QuédateEnCasa | Mira estas 𝐜𝐫𝐞𝐚𝐜𝐢𝐨𝐧𝐞𝐬 𝐨𝐫𝐢𝐠𝐢𝐧𝐚...     False

1       Contra el #Coronavirus 😷👏○□\n\n🏠 #QuédateEnCas...     False

2       @HondaMexico extiende suspensión de sus planta...     False
3       Con manos limpias, seguro estarás mejor. #Prev...     False
4       🚨😷 Baja California suma cuatro muertos por #CO...     False

...                                                   ...       ...
536130  □😷 A partir del lunes circularán menos taxis e...     False

536131  #Brasil: ¿Puede #Bolsonaro ser destituido por ...     False
536148  #UltimoMinuto Tercera muerte por #Covid19 es c...     False
536154  #Covid_19\nPermanezcamos en casa sin caer en l...      True
536155  @PaisMenosPeor Ni zombies en el fin del mundo,...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
0            False              2307             15           536720

1            False              1473              0             1847

2            False               422              0              538

3            False               349              0             1015

4            False              3640              1           293891

...            ...               ...            ...              ...

536130       False              1322              0            19522

536131       False              6753              3            21444

536148       False              3886              1            29675

536154       False                30              0               57

536155       False             26271              0              349


        friends_count account_created_at  verified lang  lang_count  \
0                1164         2010-04-15         1   es       82558
1                 252         2014-08-10         0   es       82558
```

```
2                 780         2016-09-13          0   es       82558
3                  41         2017-05-04          0   es       82558
4                 269         2009-11-10          1   es       82558
...               ...                ...       ...  ...         ...
536130            380         2010-05-11          0   es       82558
536131           1937         2009-04-08          0   es       82558
536148            126         2012-09-18          0   es       82558
536154             59         2015-10-08          0   es       82558
536155           1214         2011-09-17          0   es       82558
```

```
        follower_classification
0                             1
1                             1
2                             1
3                             1
4                             1
...                         ...
536130                        1
536131                        1
536148                        1
536154                        0
536155                        0

[82550 rows x 16 columns]
```

1.4.1.6: Investigation of Spanish Tweets Retweets and Quotes and Removing Unnecessary Columns

Again for the Spanish tweets, we only want tweets that are that individual's personal sentiment, rather than quoting another user's opinion. This is because including elements such as retweets and quotes could include multiple users that are retweeting and/or quoting the same tweet, thus again skewing overall sentiment for training.

Thus, as long as they don't account for too many tweets, we should remove all rows that are quotes or tweets.

Likewise, we should remove any columns that are not necessary for overall sentiment analysis, such as personal identifiers.

```
# Checking number of tweets that are retweets
# Seeing that there are 0, we can remove this column easily.
spanish_tweets_df[spanish_tweets_df['is_retweet'] == True].count()
```

```
status_id                   0
user_id                     0
created_at                  0
screen_name                 0
text                        0
is_quote                    0
is_retweet                  0
```

```
favourites_count            0
retweet_count               0
followers_count             0
friends_count               0
account_created_at          0
verified                    0
lang                        0
lang_count                  0
follower_classification     0
dtype: int64
```

```
# Checking number of tweets that are quotes.
# Seeing that there's a small proportion of tweets that are quotes,
they can be removed without overly affecting sentiment analysis.
spanish_tweets_df[spanish_tweets_df['is_quote'] == True].count()
```

```
status_id                   6788
user_id                     6788
created_at                  6788
screen_name                 6788
text                        6788
is_quote                    6788
is_retweet                  6788
favourites_count            6788
retweet_count               6788
followers_count             6788
friends_count               6788
account_created_at          6788
verified                    6788
lang                        6788
lang_count                  6788
follower_classification     6788
dtype: int64
```

```
# Removes all rows from the dataset that are quotes
spanish_tweets_df = spanish_tweets_df[spanish_tweets_df['is_quote'] ==
False]
spanish_tweets_df
```

```
                  status_id              user_id  created_at
screen_name  \
0       1245863586586439680            133184048  2020-04-03
EcuadorTV
1       1245863584417992704           2722502906  2020-04-03
GradaNorteMX
2       1245863584799678464  775704843994353665  2020-04-03
AutoSupplyNews
3       1245863587307868160  860252856829587457  2020-04-03
IMSS_SanLuis
4       1245863586892779523             88957440  2020-04-03
```

```
Imagen_Mx
...                      ...                  ...          ...
...
536129  1246225949525282816            29022242  2020-04-03
FernandaCamino
536130  1246225952608325632           142811824  2020-04-03
QuintaFuerzaMX
536131  1246225954210537472            29655554  2020-04-03
Servindi
536148  1246225966667628546           830542951  2020-04-03
LaSandino
536155  1246225970379530245           374870492  2020-04-03
cemgiraldo

                                                      text  is_quote  \
0       #QuédateEnCasa | Mira estas creaciones origina...     False

1       Contra el #Coronavirus 😷✊◯\n\n🏠 #QuédateEnCas...     False

2       @HondaMexico extiende suspensión de sus planta...     False
3       Con manos limpias, seguro estarás mejor. #Prev...     False
4       🚨😳 Baja California suma cuatro muertos por #CO...     False

...                                                   ...       ...
536129  "#Coronavirus en los #EstadosUnidos: tras regi...     False
536130  \n😳 A partir del lunes circularán menos taxis e...     False

536131  #Brasil: ¿Puede #Bolsonaro ser destituido por ...     False
536148  #UltimoMinuto Tercera muerte por #Covid19 es c...     False
536155  @PaisMenosPeor Ni zombies en el fin del mundo,...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
0            False              2307             15           536720

1            False              1473              0             1847

2            False               422              0              538

3            False               349              0             1015

4            False              3640              1           293891

...            ...               ...            ...              ...

536129       False             75563              0             1041

536130       False              1322              0            19522

536131       False              6753              3            21444
```

```
536148       False               3886               1             29675

536155       False              26271               0               349


        friends_count account_created_at  verified lang  lang_count  \
0                1164         2010-04-15         1   es       82558
1                 252         2014-08-10         0   es       82558
2                 780         2016-09-13         0   es       82558
3                  41         2017-05-04         0   es       82558
4                 269         2009-11-10         1   es       82558
...               ...                ...       ...  ...         ...
536129            300         2009-04-05         0   es       82558
536130            380         2010-05-11         0   es       82558
536131           1937         2009-04-08         0   es       82558
536148            126         2012-09-18         0   es       82558
536155           1214         2011-09-17         0   es       82558

        follower_classification
0                             1
1                             1
2                             1
3                             1
4                             1
...                         ...
536129                        1
536130                        1
536131                        1
536148                        1
536155                        0

[75762 rows x 16 columns]
```

```
# Removes all rows from the dataset that are retweets
spanish_tweets_df = spanish_tweets_df[spanish_tweets_df['is_retweet']
== False]
spanish_tweets_df
```

```
                  status_id             user_id  created_at
screen_name  \
0       1245863586586439680           133184048  2020-04-03
EcuadorTV
1       1245863584417992704          2722502906  2020-04-03
GradaNorteMX
2       1245863584799678464  775704843994353665  2020-04-03
AutoSupplyNews
3       1245863587307868160  860252856829587457  2020-04-03
IMSS_SanLuis
4       1245863586892779523            88957440  2020-04-03
Imagen_Mx
```

```
...                    ...                 ...         ...
...
536129  1246225949525282816            29022242  2020-04-03
FernandaCamino
536130  1246225952608325632           142811824  2020-04-03
QuintaFuerzaMX
536131  1246225954210537472            29655554  2020-04-03
Servindi
536148  1246225966667628546           830542951  2020-04-03
LaSandino
536155  1246225970379530245           374870492  2020-04-03
cemgiraldo

                                                      text  is_quote  \
0       #QuédateEnCasa | Mira estas 𝐜𝐫𝐞𝐚𝐜𝐢𝐨𝐧𝐞𝐬 𝐨𝐫𝐢𝐠𝐢𝐧𝐚...     False

1       Contra el #Coronavirus 😷👏🏻□\n\n🏠 #QuédateEnCas...     False

2       @HondaMexico extiende suspensión de sus planta...     False
3       Con manos limpias, seguro estarás mejor. #Prev...     False
4       🚨😷 Baja California suma cuatro muertos por #CO...     False

...                                                   ...       ...
536129  "#Coronavirus en los #EstadosUnidos: tras regi...     False
536130  □😷 A partir del lunes circularán menos taxis e...     False

536131  #Brasil: ¿Puede #Bolsonaro ser destituido por ...     False
536148  #UltimoMinuto Tercera muerte por #Covid19 es c...     False
536155  @PaisMenosPeor Ni zombies en el fin del mundo,...     False

        is_retweet  favourites_count  retweet_count
followers_count  \
0            False              2307             15           536720

1            False              1473              0             1847

2            False               422              0              538

3            False               349              0             1015

4            False              3640              1           293891

...            ...               ...            ...              ...

536129       False             75563              0             1041

536130       False              1322              0            19522

536131       False              6753              3            21444

536148       False              3886              1            29675
```

```
536155       False              26271               0               349


        friends_count account_created_at  verified lang  lang_count  \
0                1164         2010-04-15         1   es       82558
1                 252         2014-08-10         0   es       82558
2                 780         2016-09-13         0   es       82558
3                  41         2017-05-04         0   es       82558
4                 269         2009-11-10         1   es       82558
...               ...                ...       ...  ...         ...
536129            300         2009-04-05         0   es       82558
536130            380         2010-05-11         0   es       82558
536131           1937         2009-04-08         0   es       82558
536148            126         2012-09-18         0   es       82558
536155           1214         2011-09-17         0   es       82558

        follower_classification
0                             1
1                             1
2                             1
3                             1
4                             1
...                         ...
536129                        1
536130                        1
536131                        1
536148                        1
536155                        0

[75762 rows x 16 columns]
```

```
# As they are now unnecessary, we can remove the is_quote and
is_retweet columns now
spanish_tweets_df.drop(['is_quote', 'is_retweet'], axis=1,
inplace=True)
spanish_tweets_df
```

```
                  status_id             user_id  created_at
screen_name  \
0       1245863586586439680           133184048  2020-04-03
EcuadorTV
1       1245863584417992704          2722502906  2020-04-03
GradaNorteMX
2       1245863584799678464  775704843994353665  2020-04-03
AutoSupplyNews
3       1245863587307868160  860252856829587457  2020-04-03
IMSS_SanLuis
4       1245863586892779523            88957440  2020-04-03
Imagen_Mx
```

```
...                     ...                  ...         ...
...
536129  1246225949525282816            29022242  2020-04-03
FernandaCamino
536130  1246225952608325632           142811824  2020-04-03
QuintaFuerzaMX
536131  1246225954210537472            29655554  2020-04-03
Servindi
536148  1246225966667628546           830542951  2020-04-03
LaSandino
536155  1246225970379530245           374870492  2020-04-03
cemgiraldo

                                                      text
favourites_count  \
0       #QuédateEnCasa | Mira estas creaciones origina...
2307
1       Contra el #Coronavirus 😷👏○□\n\n🏠 #QuédateEnCas...
1473
2       @HondaMexico extiende suspensión de sus planta...
422
3       Con manos limpias, seguro estarás mejor. #Prev...
349
4       🚨😳 Baja California suma cuatro muertos por #CO...
3640
...                                                   ...
...
536129  "#Coronavirus en los #EstadosUnidos: tras regi...
75563
536130  □😳 A partir del lunes circularán menos taxis e...
1322
536131  #Brasil: ¿Puede #Bolsonaro ser destituido por ...
6753
536148  #UltimoMinuto Tercera muerte por #Covid19 es c...
3886
536155  @PaisMenosPeor Ni zombies en el fin del mundo,...
26271

        retweet_count  followers_count  friends_count
account_created_at  \
0                  15           536720           1164         2010-04-
15
1                   0             1847            252         2014-08-
10
2                   0              538            780         2016-09-
13
3                   0             1015             41         2017-05-
04
4                   1           293891            269         2009-11-
```

```
10
...                ...               ...             ...                 .
..
536129               0             1041             300          2009-04-
05
536130               0            19522             380          2010-05-
11
536131               3            21444            1937          2009-04-
08
536148               1            29675             126          2012-09-
18
536155               0              349            1214          2011-09-
17

        verified lang  lang_count  follower_classification
0              1   es       82558                        1
1              0   es       82558                        1
2              0   es       82558                        1
3              0   es       82558                        1
4              1   es       82558                        1
...          ...  ...         ...                      ...
536129         0   es       82558                        1
536130         0   es       82558                        1
536131         0   es       82558                        1
536148         0   es       82558                        1
536155         0   es       82558                        0

[75762 rows x 14 columns]
```

```
# Looking at the status_id, user_id, and screen_name columns, we don't
need them for Spanish tweet sentiment analysis, so we can drop them.
spanish_tweets_df.drop(['status_id', 'user_id', 'screen_name'],
axis=1, inplace=True)
spanish_tweets_df
```

```
        created_at                                               text
\
0       2020-04-03  #QuédateEnCasa | Mira estas creaciones
origina...
1       2020-04-03  Contra el #Coronavirus 😷👐◯\n\n🏠
#QuédateEnCas...
2       2020-04-03  @HondaMexico extiende suspensión de sus planta...

3       2020-04-03  Con manos limpias, seguro estarás mejor. #Prev...

4       2020-04-03  🚨😷 Baja California suma cuatro muertos por #CO...

...            ...                                                ...

536129  2020-04-03  "#Coronavirus en los #EstadosUnidos: tras regi...
```

```
536130  2020-04-03  🤔 A partir del lunes circularán menos taxis e...
536131  2020-04-03  #Brasil: ¿Puede #Bolsonaro ser destituido por ...
536148  2020-04-03  #UltimoMinuto Tercera muerte por #Covid19 es c...
536155  2020-04-03  @PaisMenosPeor Ni zombies en el fin del mundo,...

        favourites_count  retweet_count  followers_count
friends_count  \
0                   2307             15           536720
1164
1                   1473              0             1847
252
2                    422              0              538
780
3                    349              0             1015
41
4                   3640              1           293891
269
...                  ...            ...              ...             ..
.
536129             75563              0             1041
300
536130              1322              0            19522
380
536131              6753              3            21444
1937
536148              3886              1            29675
126
536155             26271              0              349
1214

       account_created_at  verified lang  lang_count
follower_classification
0              2010-04-15         1   es       82558
1
1              2014-08-10         0   es       82558
1
2              2016-09-13         0   es       82558
1
3              2017-05-04         0   es       82558
1
4              2009-11-10         1   es       82558
1
...                   ...       ...  ...         ...
...
536129         2009-04-05         0   es       82558
```

```
1
536130          2010-05-11          0   es       82558
1
536131          2009-04-08          0   es       82558
1
536148          2012-09-18          0   es       82558
1
536155          2011-09-17          0   es       82558
0

[75762 rows x 11 columns]
```

```
# Furthermore, as we know this is the Spanish tweet dataset, and don't
need lang_count anymore, we can remove those as well
spanish_tweets_df.drop(['lang', 'lang_count'], axis=1, inplace=True)
spanish_tweets_df
```

```
        created_at                                                  text
\
0       2020-04-03  #QuédateEnCasa | Mira estas 𝐜𝐫𝐞𝐚𝐜𝐢𝐨𝐧𝐞𝐬
𝐨𝐫𝐢𝐠𝐢𝐧𝐚...
1       2020-04-03  Contra el #Coronavirus 😷👏◯□\n\n🏠
#QuédateEnCas...
2       2020-04-03  @HondaMexico extiende suspensión de sus planta...

3       2020-04-03  Con manos limpias, seguro estarás mejor. #Prev...

4       2020-04-03  🚨😳 Baja California suma cuatro muertos por #CO...

...            ...                                                 ...

536129  2020-04-03  "#Coronavirus en los #EstadosUnidos: tras regi...

536130  2020-04-03  □😳 A partir del lunes circularán menos taxis e...

536131  2020-04-03  #Brasil: ¿Puede #Bolsonaro ser destituido por ...

536148  2020-04-03  #UltimoMinuto Tercera muerte por #Covid19 es c...

536155  2020-04-03  @PaisMenosPeor Ni zombies en el fin del mundo,...


        favourites_count  retweet_count  followers_count
friends_count  \
0                   2307             15           536720
1164
1                   1473              0             1847
252
2                    422              0              538
780
3                    349              0             1015
```

```
41
4                    3640               1           293891
269
...                   ...             ...              ...             ..
.
536129              75563               0             1041
300
536130               1322               0            19522
380
536131               6753               3            21444
1937
536148               3886               1            29675
126
536155              26271               0              349
1214

       account_created_at  verified  follower_classification
0              2010-04-15         1                        1
1              2014-08-10         0                        1
2              2016-09-13         0                        1
3              2017-05-04         0                        1
4              2009-11-10         1                        1
...                   ...       ...                      ...
536129         2009-04-05         0                        1
536130         2010-05-11         0                        1
536131         2009-04-08         0                        1
536148         2012-09-18         0                        1
536155         2011-09-17         0                        0

[75762 rows x 9 columns]
```

1.4.2: Spanish Tweets Sentiment Analysis

1.4.2.1: Spanish Sentiment Analysis Preprocessing

Uses the NLTK library, specifically the Spanish parts of it.

We have to again first tokenize each tweet, before turning them into lowercase, stemming them with the SnowballStemmer library, and removing any non-alphabetic characters or "stopwords" according to the Spanish library.

Because the SnowballStemmer library is an improvement over the normal PorterStemmer library, and because it is included in the NLTK library and supports multi-lingual analysis for languages like Spanish, we have chosen to use that to increase the accuracy of our sentiment analysis for Spanish tweets.

```
# Imports the Spanish stopwords set from NLTK
from nltk.corpus import stopwords
```

```
nltk.download('stopwords')
spanish_stopwords = set(stopwords.words('spanish'))
```

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

```
# Imports the Spanish stemmer toolkit from SnowballStemmer
from nltk.stem.snowball import SnowballStemmer
spanish_snowball_stemmer = SnowballStemmer(language='spanish')

# Creates a user-defined function that runs the processing pipeline of
lowercasing/tokenization/stemming/removing stopwords
# Returns processed string
@numba.jit
def tokenize_spanish_content(content):
  tokens = nltk.word_tokenize(content, language='spanish')
  final_string = []
  for tk in tokens:
    lowercase = tk.lower()
    #spanish_snowball_stemmer.stem(lowercase)
    stemmed_lowercase = spanish_snowball_stemmer.stem(lowercase)
    if ((str.isalpha(stemmed_lowercase)) and (stemmed_lowercase not in
spanish_stopwords)):
      final_string.append(stemmed_lowercase)
    else:
      continue

  return final_string

# Creates a processed_text column from running the user-defined
function on each tweet in spanish_tweets_df
spanish_tweets_df['processed_text'] =
spanish_tweets_df['text'].map(lambda x: tokenize_spanish_content(x))
spanish_tweets_df
```

```
<ipython-input-112-65d167fc35be>:3: NumbaWarning:
Compilation is falling back to object mode WITH looplifting enabled
because Function "tokenize_spanish_content" failed type inference due
to: Untyped global name 'spanish_snowball_stemmer': Cannot determine
Numba type of <class 'nltk.stem.snowball.SnowballStemmer'>

File "<ipython-input-112-65d167fc35be>", line 10:
def tokenize_spanish_content(content):
    <source elided>
    #spanish_snowball_stemmer.stem(lowercase)
    stemmed_lowercase = spanish_snowball_stemmer.stem(lowercase)
    ^

  @numba.jit
<ipython-input-112-65d167fc35be>:3: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
```

```
because Function "tokenize_spanish_content" failed type inference due
to: Cannot determine Numba type of <class
'numba.core.dispatcher.LiftedLoop'>

File "<ipython-input-112-65d167fc35be>", line 7:
def tokenize_spanish_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "tokenize_spanish_content" was compiled
in object mode without forceobj=True, but has lifted loops.

File "<ipython-input-112-65d167fc35be>", line 5:
def tokenize_spanish_content(content):
  tokens = nltk.word_tokenize(content, language='spanish')
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-112-65d167fc35be>", line 5:
def tokenize_spanish_content(content):
  tokens = nltk.word_tokenize(content, language='spanish')
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,
<ipython-input-112-65d167fc35be>:3: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "tokenize_spanish_content" failed type inference due
to: Untyped global name 'spanish_snowball_stemmer': Cannot determine
Numba type of <class 'nltk.stem.snowball.SnowballStemmer'>

File "<ipython-input-112-65d167fc35be>", line 10:
def tokenize_spanish_content(content):
    <source elided>
    #spanish_snowball_stemmer.stem(lowercase)
    stemmed_lowercase = spanish_snowball_stemmer.stem(lowercase)
    ^
```

```
  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "tokenize_spanish_content" was compiled
in object mode without forceobj=True.

File "<ipython-input-112-65d167fc35be>", line 7:
def tokenize_spanish_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-112-65d167fc35be>", line 7:
def tokenize_spanish_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,

        created_at                                                 text
\
0       2020-04-03  #QuédateEnCasa | Mira estas creaciones
origina...
1       2020-04-03  Contra el #Coronavirus 😷👏🏻□\n\n🏠
#QuédateEnCas...
2       2020-04-03  @HondaMexico extiende suspensión de sus planta...

3       2020-04-03  Con manos limpias, seguro estarás mejor. #Prev...

4       2020-04-03  🤴😷 Baja California suma cuatro muertos por #CO...

...            ...                                                 ...

536129  2020-04-03  "#Coronavirus en los #EstadosUnidos: tras regi...

536130  2020-04-03  □😷 A partir del lunes circularán menos taxis e...

536131  2020-04-03  #Brasil: ¿Puede #Bolsonaro ser destituido por ...
```

```
536148  2020-04-03  #UltimoMinuto Tercera muerte por #Covid19 es c...

536155  2020-04-03  @PaisMenosPeor Ni zombies en el fin del mundo,...


        favourites_count  retweet_count  followers_count
friends_count  \
0                   2307             15           536720
1164
1                   1473              0             1847
252
2                    422              0              538
780
3                    349              0             1015
41
4                   3640              1           293891
269
...                  ...            ...              ...             ..
.
536129             75563              0             1041
300
536130              1322              0            19522
380
536131              6753              3            21444
1937
536148              3886              1            29675
126
536155             26271              0              349
1214

       account_created_at  verified  follower_classification  \
0              2010-04-15         1                         1
1              2014-08-10         0                         1
2              2016-09-13         0                         1
3              2017-05-04         0                         1
4              2009-11-10         1                         1
...                   ...       ...                       ...
536129         2009-04-05         0                         1
536130         2010-05-11         0                         1
536131         2009-04-08         0                         1
536148         2012-09-18         0                         1
536155         2011-09-17         0                         0

                                          processed_text
0       [quedateencas, mir, creaciones, originales, pa...
1       [contr, coronavirus, quedateencas, https, grad...
2       [hondamex, extiend, suspension, plant, canad, ...
3       [man, limpi, segur, estaras, mejor, prevencion...
4       [baj, californi, sum, cuatr, muert, https, https]
...                                                   ...
```

```
536129  [coronavirus, estadosun, tras, registr, pic, m...
536130  [part, lun, circul, men, taxis, call, playadel...
536131  [brasil, bolsonar, ser, destitu, manej, jos, d...
536148  [ultimominut, tercer, muert, confirm, autor, e...
536155  [paismenospeor, zombi, fin, mund, eeuu, salv, ...

[75762 rows x 10 columns]
```

```
# A full similar user-defined function to return a cleaned, but not
split into tokens version of the tweet for model training purposes
@numba.jit
def clean_tweet_spanish(content):
  final_string = ""
  tokens = nltk.word_tokenize(content)
  for word in tokens:
    stemmed = word.lower()
    if(stemmed not in spanish_stopwords):
      for char in range(0, len(stemmed)):
        stemmed = stemmed.replace('#', '')
        stemmed = stemmed.replace('@', '')
      final_string = final_string + stemmed + " "
    else:
      continue
  return final_string

# Applying that user-defined function to the text column to create a
new training_text column
spanish_tweets_df['training_text'] =
spanish_tweets_df['text'].apply(lambda x: clean_tweet_spanish(x))
```

```
<ipython-input-114-2cb44a4883a3>:2: NumbaWarning:
Compilation is falling back to object mode WITH looplifting enabled
because Function "clean_tweet_spanish" failed type inference due to:
Unknown attribute 'word_tokenize' of type Module(<module 'nltk' from
'/usr/local/lib/python3.8/dist-packages/nltk/__init__.py'>)

File "<ipython-input-114-2cb44a4883a3>", line 5:
def clean_tweet_spanish(content):
    <source elided>
  final_string = ""
  tokens = nltk.word_tokenize(content)
  ^

During: typing of get attribute at <ipython-input-114-2cb44a4883a3>
(5)

File "<ipython-input-114-2cb44a4883a3>", line 5:
def clean_tweet_spanish(content):
    <source elided>
  final_string = ""
```

```
  tokens = nltk.word_tokenize(content)
  ^

  @numba.jit
<ipython-input-114-2cb44a4883a3>:2: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "clean_tweet_spanish" failed type inference due to:
Cannot determine Numba type of <class
'numba.core.dispatcher.LiftedLoop'>

File "<ipython-input-114-2cb44a4883a3>", line 6:
def clean_tweet_spanish(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "clean_tweet_spanish" was compiled in
object mode without forceobj=True, but has lifted loops.

File "<ipython-input-114-2cb44a4883a3>", line 4:
def clean_tweet_spanish(content):
  final_string = ""
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-114-2cb44a4883a3>", line 4:
def clean_tweet_spanish(content):
  final_string = ""
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "clean_tweet_spanish" was compiled in
object mode without forceobj=True.

File "<ipython-input-114-2cb44a4883a3>", line 6:
def clean_tweet_spanish(content):
    <source elided>
```

```
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-114-2cb44a4883a3>", line 6:
def clean_tweet_spanish(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,
```

1.4.2.2: Spanish Sentiment Analysis Calculation Using sentiment-spanish

For our Spanish tweet sentiment analysis calculation, we will be using the sentiment-spanish library. (https://github.com/sentiment-analysis-spanish/sentiment-spanish)

sentiment-analysis-spanish is a sentiment analysis and natural language processing library trained on Spanish-language user reviews of various sites such as eBay, TripAdvisor, and more that has been used to study sentiment in multiple academic studies.

For example, it has been used to investigate Twitter fake news spreaders (https://par.nsf.gov/servlets/purl/10230411), combating cyberterrorism (https://repository.urosario.edu.co/handle/10336/34736), and importantly, an analysis on overall COVID-19 pandemic-era sentiment on Twitter, similar to the process that we are implementing (https://link.springer.com/article/10.1007/s13278-021-00825-0#Sec9).

Thus, we felt that for the purposes of our analysis, which is to investigate multilingual sentiment in tweets regarding COVID-19, that the sentiment-analysis-spanish library would be a good fit. Because of its usage by multiple studies in reputable journals, including one study that did exactly what we are trying to investigate, and its origin as a model trained on tweet data fitting our current analysis, this toolkit was a good fit for what we wished to do.

```
# Installs the sentiment-analysis-spanish library
!pip install sentiment-analysis-spanish

Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Collecting sentiment-analysis-spanish
```

```
  Downloading sentiment_analysis_spanish-0.0.25-py3-none-any.whl (30.0
MB)
ent-analysis-spanish
Successfully installed sentiment-analysis-spanish-0.0.25

# Installs keras tensorflow, which is needed for sentiment-analysis-
library
!pip install keras tensorflow

Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: keras in /usr/local/lib/python3.8/dist-
packages (2.9.0)
Requirement already satisfied: tensorflow in
/usr/local/lib/python3.8/dist-packages (2.9.2)
Requirement already satisfied: flatbuffers<2,>=1.12 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.12)
Requirement already satisfied: tensorflow-estimator<2.10.0,>=2.9.0rc0
in /usr/local/lib/python3.8/dist-packages (from tensorflow) (2.9.0)
Requirement already satisfied: wrapt>=1.11.0 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.14.1)
Requirement already satisfied: gast<=0.4.0,>=0.2.1 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (0.4.0)
Requirement already satisfied: numpy>=1.20 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.21.6)
Requirement already satisfied: absl-py>=1.0.0 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.3.0)
Requirement already satisfied: keras-preprocessing>=1.1.1 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.1.2)
Requirement already satisfied: six>=1.12.0 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.15.0)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.51.1)
Requirement already satisfied: libclang>=13.0.0 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (14.0.6)
Requirement already satisfied: google-pasta>=0.1.1 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (0.2.0)
Requirement already satisfied: typing-extensions>=3.6.6 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (4.4.0)
Requirement already satisfied: protobuf<3.20,>=3.9.2 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (3.19.6)
Requirement already satisfied: packaging in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (21.3)
Requirement already satisfied: termcolor>=1.1.0 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (2.1.1)
Requirement already satisfied: h5py>=2.9.0 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (3.1.0)
Requirement already satisfied: setuptools in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (57.4.0)
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in
```

```
/usr/local/lib/python3.8/dist-packages (from tensorflow) (0.28.0)
Requirement already satisfied: tensorboard<2.10,>=2.9 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (2.9.1)
Requirement already satisfied: opt-einsum>=2.3.2 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (3.3.0)
Requirement already satisfied: astunparse>=1.6.0 in
/usr/local/lib/python3.8/dist-packages (from tensorflow) (1.6.3)
Requirement already satisfied: wheel<1.0,>=0.23.0 in
/usr/local/lib/python3.8/dist-packages (from astunparse>=1.6.0-
>tensorflow) (0.38.4)
Requirement already satisfied: markdown>=2.6.8 in
/usr/local/lib/python3.8/dist-packages (from tensorboard<2.10,>=2.9-
>tensorflow) (3.4.1)
Requirement already satisfied: google-auth<3,>=1.6.3 in
/usr/local/lib/python3.8/dist-packages (from tensorboard<2.10,>=2.9-
>tensorflow) (2.15.0)
Requirement already satisfied: google-auth-oauthlib<0.5,>=0.4.1 in
/usr/local/lib/python3.8/dist-packages (from tensorboard<2.10,>=2.9-
>tensorflow) (0.4.6)
Requirement already satisfied: requests<3,>=2.21.0 in
/usr/local/lib/python3.8/dist-packages (from tensorboard<2.10,>=2.9-
>tensorflow) (2.23.0)
Requirement already satisfied: tensorboard-plugin-wit>=1.6.0 in
/usr/local/lib/python3.8/dist-packages (from tensorboard<2.10,>=2.9-
>tensorflow) (1.8.1)
Requirement already satisfied: werkzeug>=1.0.1 in
/usr/local/lib/python3.8/dist-packages (from tensorboard<2.10,>=2.9-
>tensorflow) (1.0.1)
Requirement already satisfied: tensorboard-data-server<0.7.0,>=0.6.0
in /usr/local/lib/python3.8/dist-packages (from
tensorboard<2.10,>=2.9->tensorflow) (0.6.1)
Requirement already satisfied: rsa<5,>=3.1.4 in
/usr/local/lib/python3.8/dist-packages (from google-auth<3,>=1.6.3-
>tensorboard<2.10,>=2.9->tensorflow) (4.9)
Requirement already satisfied: pyasn1-modules>=0.2.1 in
/usr/local/lib/python3.8/dist-packages (from google-auth<3,>=1.6.3-
>tensorboard<2.10,>=2.9->tensorflow) (0.2.8)
Requirement already satisfied: cachetools<6.0,>=2.0.0 in
/usr/local/lib/python3.8/dist-packages (from google-auth<3,>=1.6.3-
>tensorboard<2.10,>=2.9->tensorflow) (5.2.0)
Requirement already satisfied: requests-oauthlib>=0.7.0 in
/usr/local/lib/python3.8/dist-packages (from google-auth-
oauthlib<0.5,>=0.4.1->tensorboard<2.10,>=2.9->tensorflow) (1.3.1)
Requirement already satisfied: importlib-metadata>=4.4 in
/usr/local/lib/python3.8/dist-packages (from markdown>=2.6.8-
>tensorboard<2.10,>=2.9->tensorflow) (4.13.0)
Requirement already satisfied: zipp>=0.5 in
/usr/local/lib/python3.8/dist-packages (from importlib-metadata>=4.4-
>markdown>=2.6.8->tensorboard<2.10,>=2.9->tensorflow) (3.11.0)
```

```
Requirement already satisfied: pyasn1<0.5.0,>=0.4.6 in
/usr/local/lib/python3.8/dist-packages (from pyasn1-modules>=0.2.1-
>google-auth<3,>=1.6.3->tensorboard<2.10,>=2.9->tensorflow) (0.4.8)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.8/dist-packages (from requests<3,>=2.21.0-
>tensorboard<2.10,>=2.9->tensorflow) (2022.9.24)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1
in /usr/local/lib/python3.8/dist-packages (from requests<3,>=2.21.0-
>tensorboard<2.10,>=2.9->tensorflow) (1.24.3)
Requirement already satisfied: idna<3,>=2.5 in
/usr/local/lib/python3.8/dist-packages (from requests<3,>=2.21.0-
>tensorboard<2.10,>=2.9->tensorflow) (2.10)
Requirement already satisfied: chardet<4,>=3.0.2 in
/usr/local/lib/python3.8/dist-packages (from requests<3,>=2.21.0-
>tensorboard<2.10,>=2.9->tensorflow) (3.0.4)
Requirement already satisfied: oauthlib>=3.0.0 in
/usr/local/lib/python3.8/dist-packages (from requests-oauthlib>=0.7.0-
>google-auth-oauthlib<0.5,>=0.4.1->tensorboard<2.10,>=2.9->tensorflow)
(3.2.2)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/usr/local/lib/python3.8/dist-packages (from packaging->tensorflow)
(3.0.9)

# Imports the sentiment-analysis class from sentiment-analysis-spanish
from sentiment_analysis_spanish import sentiment_analysis

# Creates an instance of the sentiment_analysis class, and defines a
user-defined function for finding sentiment
spanish_sa = sentiment_analysis.SentimentAnalysisSpanish()

def retrieve_spanish_sentiment(content):
  sentence = ' '.join(word for word in content)
  return spanish_sa.sentiment(sentence)

/usr/local/lib/python3.8/dist-packages/sklearn/base.py:329:
UserWarning: Trying to unpickle estimator CountVectorizer from version
0.23.2 when using version 1.0.2. This might lead to breaking code or
invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/modules/model_persistence.html#securit
y-maintainability-limitations
  warnings.warn(
/usr/local/lib/python3.8/dist-packages/sklearn/base.py:329:
UserWarning: Trying to unpickle estimator MultinomialNB from version
0.23.2 when using version 1.0.2. This might lead to breaking code or
invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/modules/model_persistence.html#securit
y-maintainability-limitations
  warnings.warn(
```

```
# Runs the sentiment analyzer on every processed tweet to get a
sentiment score
spanish_tweets_df['sentiment'] =
spanish_tweets_df['processed_text'].apply(lambda x:
retrieve_spanish_sentiment(x))
spanish_tweets_df
```

```
        created_at                                                text
\
0       2020-04-03  #QuédateEnCasa | Mira estas 𝐜𝐫𝐞𝐚𝐜𝐢𝐨𝐧𝐞𝐬
𝐨𝐫𝐢𝐠𝐢𝐧𝐚...
1       2020-04-03  Contra el #Coronavirus 😷👏🏻□\n\n🏠
#QuédateEnCas...
2       2020-04-03  @HondaMexico extiende suspensión de sus planta...

3       2020-04-03  Con manos limpias, seguro estarás mejor. #Prev...

4       2020-04-03  🚨😷 Baja California suma cuatro muertos por #CO...

...            ...                                                ...

536129  2020-04-03  "#Coronavirus en los #EstadosUnidos: tras regi...

536130  2020-04-03  □😷 A partir del lunes circularán menos taxis e...

536131  2020-04-03  #Brasil: ¿Puede #Bolsonaro ser destituido por ...

536148  2020-04-03  #UltimoMinuto Tercera muerte por #Covid19 es c...

536155  2020-04-03  @PaisMenosPeor Ni zombies en el fin del mundo,...


        favourites_count  retweet_count  followers_count
friends_count  \
0                   2307             15           536720
1164
1                   1473              0             1847
252
2                    422              0              538
780
3                    349              0             1015
41
4                   3640              1           293891
269
...                  ...            ...              ...             ..
.
536129             75563              0             1041
300
536130              1322              0            19522
380
```

```
536131              6753              3            21444
1937
536148              3886              1            29675
126
536155             26271              0              349
1214

       account_created_at  verified  follower_classification  \
0              2010-04-15         1                        1
1              2014-08-10         0                        1
2              2016-09-13         0                        1
3              2017-05-04         0                        1
4              2009-11-10         1                        1
...                   ...       ...                      ...
536129         2009-04-05         0                        1
536130         2010-05-11         0                        1
536131         2009-04-08         0                        1
536148         2012-09-18         0                        1
536155         2011-09-17         0                        0

                                          processed_text  \
0       [quedateencas, mir, creaciones, originales, pa...
1       [contr, coronavirus, quedateencas, https, grad...
2       [hondamex, extiend, suspension, plant, canad, ...
3       [man, limpi, segur, estaras, mejor, prevencion...
4       [baj, californi, sum, cuatr, muert, https, https]
...                                                   ...
536129  [coronavirus, estadosun, tras, registr, pic, m...
536130  [part, lun, circul, men, taxis, call, playadel...
536131  [brasil, bolsonar, ser, destitu, manej, jos, d...
536148  [ultimominut, tercer, muert, confirm, autor, e...
536155  [paismenospeor, zombi, fin, mund, eeuu, salv, ...

                                           training_text  sentiment
0        quédateencasa | mira creaciones originales pr...   0.464011

1        coronavirus 😔✊🏻🏠  quédateencasa 💪🏻 😷 https ...
0.497892
2        hondamexico extiende suspensión plantas  cana...   0.269797
3       manos limpias , seguro mejor .  prevenciónconcoro...   0.617922
4       🚨😷 baja california suma cuatro muertos  covid1...   0.497892

...                                                   ...        ...
536129  ``  coronavirus  estadosunidos : tras registra...   0.149585
536130  🏠😷 partir lunes circularán menos taxis calles ...   0.664921

536131   brasil : ¿puede  bolsonaro ser destituido man...   0.101106
536148   ultimominuto tercera muerte  covid19 confirma...   0.497892
536155   paismenospeor zombies fin mundo , eeuu salvan...   0.584853
```

```
[75762 rows x 12 columns]
```

The sentiment-analyzer-spanish library, when returning sentiment scores, returns them from a range of 0 to 1. Thus, we have to scale the end result to be from -1 to 1, so as to match the sentiment range that is being expressed in our French and English libraries. We do this with the simple formula of (sentiment * 2) - 1, as that should properly scale the sentiment accordingly.

```
#English and French sentiment is calculated on a scale of -1 to 1,
where as Spanish is on 0 to 1.
#Thus, we must scale Spanish sentiment so we can better compare across
langauges
spanish_tweets_df['sentiment'] =
spanish_tweets_df['sentiment'].apply(lambda x: (x*2)-1)

# We want to find the numerical columns in the data, barring
followers_count, as that has been replaced with
follower_classification
spanish_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 75762 entries, 0 to 536155
Data columns (total 12 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   created_at               75762 non-null  object
 1   text                     75762 non-null  object
 2   favourites_count         75762 non-null  int64
 3   retweet_count            75762 non-null  int64
 4   followers_count          75762 non-null  int64
 5   friends_count            75762 non-null  int64
 6   account_created_at       75762 non-null  object
 7   verified                 75762 non-null  int64
 8   follower_classification  75762 non-null  int64
 9   processed_text           75762 non-null  object
 10  training_text            75762 non-null  object
 11  sentiment                75762 non-null  float64
dtypes: float64(1), int64(6), object(5)
memory usage: 7.5+ MB
```

```
spanish_numerics_df = spanish_tweets_df[['favourites_count',
'retweet_count', 'verified',
                                         'follower_classification',
'friends_count', 'sentiment']]
spanish_numerics_df
```

```
        favourites_count  retweet_count  verified
follower_classification  \
0                   2307             15         1
1
```

```
1                    1473                0          0
1
2                     422                0          0
1
3                     349                0          0
1
4                    3640                1          1
1
...                   ...              ...        ...
...
536129              75563                0          0
1
536130               1322                0          0
1
536131               6753                3          0
1
536148               3886                1          0
1
536155              26271                0          0
0

        friends_count  sentiment
0                1164  -0.071978
1                 252  -0.004215
2                 780  -0.460407
3                  41   0.235844
4                 269  -0.004215
...               ...        ...
536129            300  -0.700830
536130            380   0.329841
536131           1937  -0.797787
536148            126  -0.004215
536155           1214   0.169705

[75762 rows x 6 columns]
```

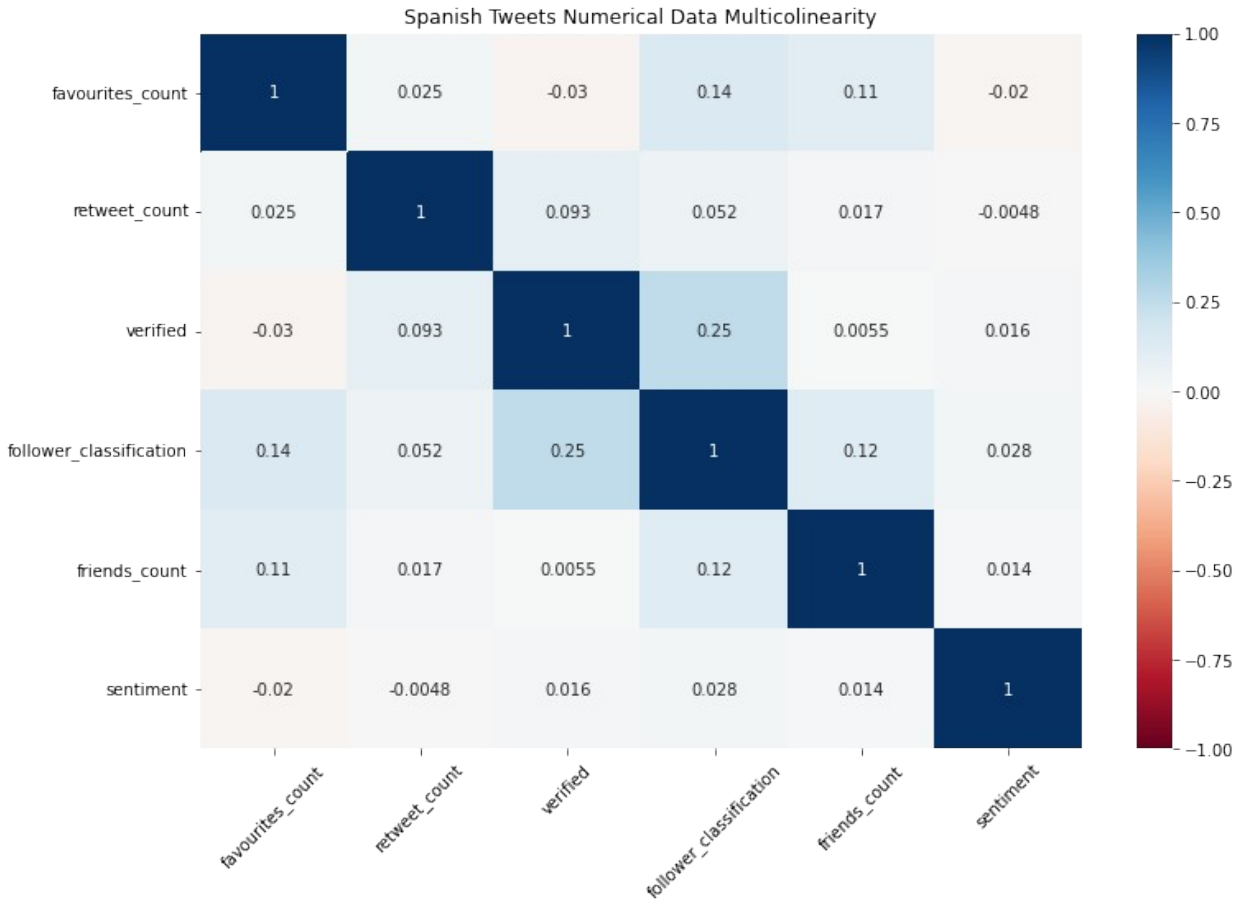
1.4.3: Spanish Tweets Visualizations

We want to visualize all of the data within the Spanish tweets dataset, so that we can find trends in our data, investigate multicolinearity, overall sentiment, etc.

```
# This is a correlation matrix that shows a heatmap of the
multicolinearity between the different variables
# As the colinearity between all of the variables isn't that high, we
don't have to remove any of them from our analysis
plt.figure(figsize=(12, 8))
spanish_corr_matrix = sns.heatmap(spanish_numerics_df.corr(), vmin=-1,
vmax=1, cmap='RdBu', annot=True)
```

```
plt.title('Spanish Tweets Numerical Data Multicolinearity')
plt.xticks(rotation=45)
```

```
(array([0.5, 1.5, 2.5, 3.5, 4.5, 5.5]),
 <a list of 6 Text major ticklabel objects>)
```



Looking at the correlation heatmap, we see that none of the variables are that highly colinear, so we don't have to remove any of them from the modeling or the analysis.

Next we will create a word cloud to get a sense of which words were most common amongst Spanish Tweets.

```
# Create the top tokens from our tokenized text. Due to the way our
tokenizer processed Spanish,
# "https" is treated as a token and became the most popular token. To
offset this, we only count
# tokens if they're not equal to "https"
top_tokens_list_spanish = spanish_tweets_df['processed_text']
top_tokens_spanish = []
for sublist in top_tokens_list_spanish:
    for element in sublist:
```

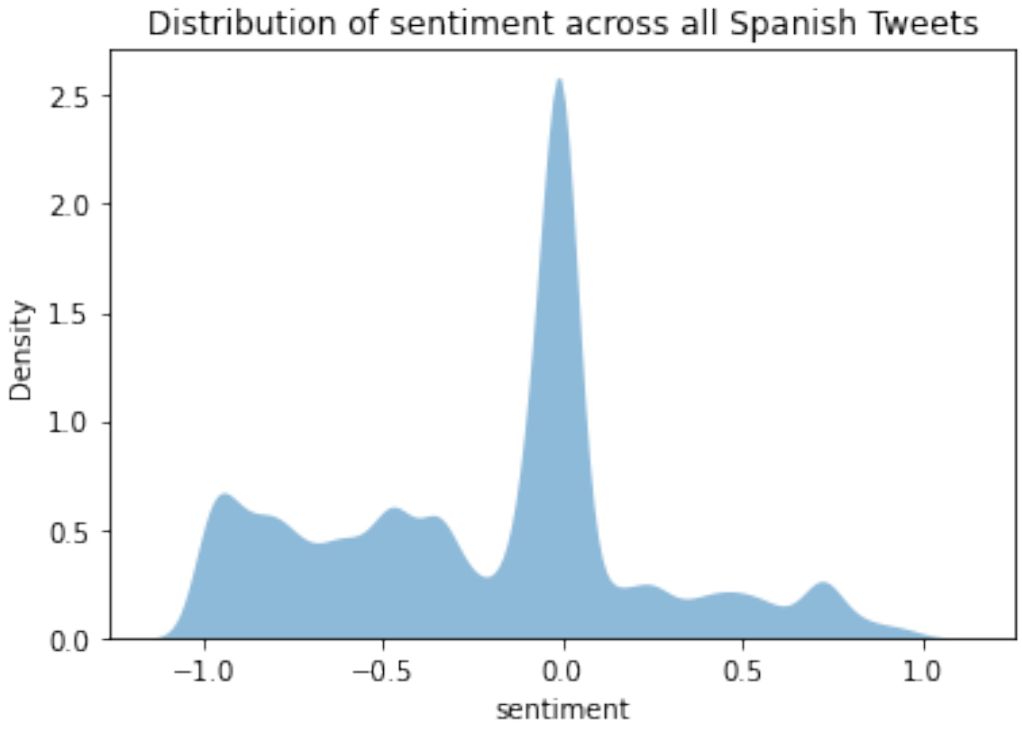
```
      if(element != 'https'):
        top_tokens_spanish.append(element)

#Count each token and collect the top 20 most frequent ones
from collections import Counter
cnt = Counter()
for word in top_tokens_spanish:
    cnt[word] += 1
top_most_common_spanish = cnt.most_common(20)

#Plot the word cloud
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
w = WordCloud(background_color='white')

cnt2 = Counter()
for word in top_tokens_spanish:
    cnt2[word] += 1

plt.figure(figsize=(12, 8))
w.generate_from_frequencies(cnt2)
plt.title('Wordcloud of Most Frequently Seen Words in Spanish Tweets')
plt.imshow(w)
```

```
<matplotlib.image.AxesImage at 0x7f582c49ddc0>
```

As expected, "coronavirus" stands out as one of the most common words along with "cuarenten" and "pandemi". Due to the nature of our tokenizer, it is difficult to tell whether the tokens appearing here are nouns and adjectives or grammatical helpers.

```
#KDE of sentiment distribution
sns.kdeplot(data=spanish_numerics_df, x='sentiment',
            fill=True, alpha=0.5, linewidth=0).set(
                title='Distribution of sentiment across all Spanish
Tweets'
            )
```

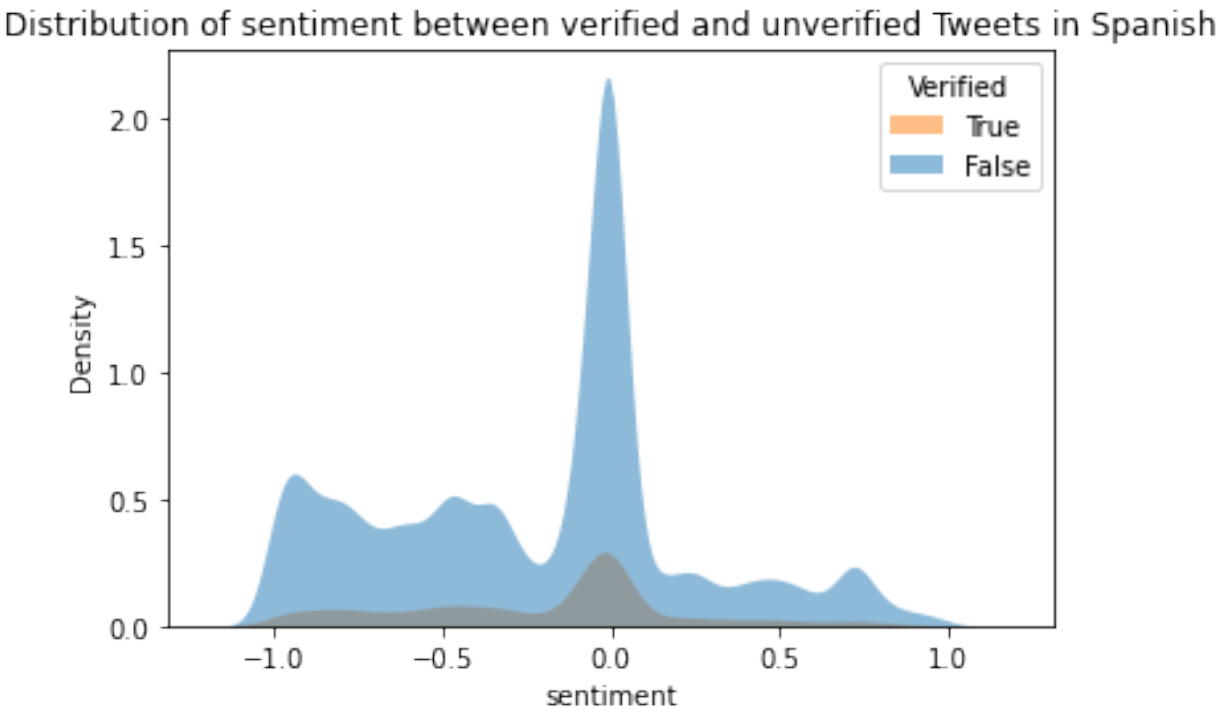
```
[Text(0.5, 1.0, 'Distribution of sentiment across all Spanish
Tweets')]
```



In this KDE plot we see that most tweets appear to have a neutral or negative sentiment, with a significant distribution appearing at the negative extreme.

```
sns.kdeplot(data = spanish_numerics_df, x='sentiment', hue='verified',

            fill=True, alpha=0.5, linewidth=0
            ).set(title='Distribution of sentiment between verified
and unverified Tweets in Spanish')
plt.legend(title='Verified', labels=['True', 'False'])
```

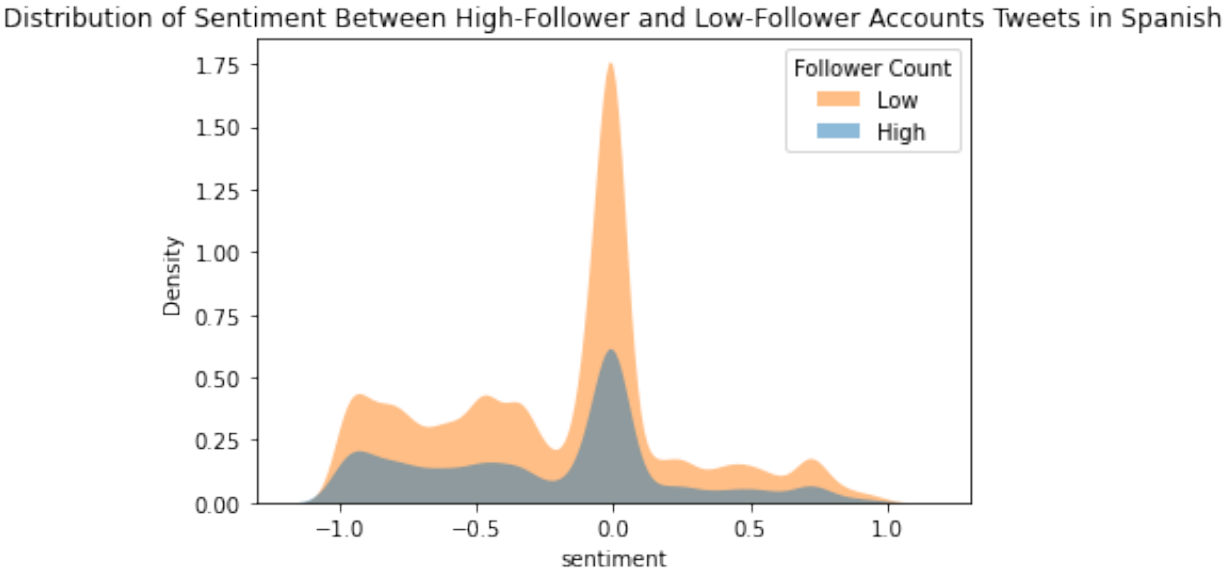
```
<matplotlib.legend.Legend at 0x7f583348a550>
```



The verified tweets reflect the overall distribution in that there are more negative tweets than positive, however the quantity of negative verified tweets in proportion to all verified tweets seems less than its unverified counterpart.

```
sns.kdeplot(data = spanish_numerics_df, x='sentiment',
hue='follower_classification',
            fill=True, alpha=0.5, linewidth=0
            ).set(title='Distribution of Sentiment Between High-
Follower and Low-Follower Accounts Tweets in Spanish')
plt.legend(title='Follower Count', labels=['Low', 'High'])
```

```
<matplotlib.legend.Legend at 0x7f58335f7280>
```



The two distributions here mirror each other, suggesting once again that there are no trends to be seen in this high following and low following demographic.

1.5: French Tweets Analysis

1.5.1: French Tweets Pre-processing and EDA

In a similar fashion to how we investigated and removed unnecessary columns and nulls from the English and Spanish tweet dataframes, we now want to do the same to the French tweets using the dataframe we created earlier, french_tweets_df.

The steps used to process the French tweets dataframe are mostly similar to that of the English and Spanish tweets, albeit with differences, mainly in the sentiment analysis area.

```
# Prints the head of 20 rows to see how the French tweets dataframe
looks like
french_tweets_df.head(20)
```

	status_id	user_id	created_at	\
141	1245863595323191296	1082203787962523648	2020-04-03T00:00:02Z	
172	1245863602633838603	57732899	2020-04-03T00:00:04Z	
212	1245863621332045827	1643982751	2020-04-03T00:00:09Z	
331	1245863683260944384	313192351	2020-04-03T00:00:23Z	
580	1245863827016519680	566248305	2020-04-03T00:00:58Z	
582	1245863830292291587	1241135805957124102	2020-04-03T00:00:58Z	
681	1245863919647780864	499403277	2020-04-03T00:01:20Z	
720	1245863955924267013	16014404	2020-04-03T00:01:28Z	
730	1245863958512205825	795045814737498112	2020-04-03T00:01:29Z	
749	1245863970201677827	797887750871666689	2020-04-03T00:01:32Z	
752	1245863976501575680	2752784274	2020-04-03T00:01:33Z	
754	1245863975541018629	560195893	2020-04-03T00:01:33Z	

```
765  1245863983430520834            352485834  2020-04-03T00:01:35Z
829  1245864035704131584  1146475150562529280  2020-04-03T00:01:47Z
838  1245864036811456513             16014404  2020-04-03T00:01:48Z
903  1245864087940194309            238080747  2020-04-03T00:02:00Z
935  1245864112506068994            400334965  2020-04-03T00:02:06Z
965  1245864140112986113           1558500486  2020-04-03T00:02:12Z
983  1245864163814981632  1120744567773585408  2020-04-03T00:02:18Z
990  1245864169384972288  1204080681552433157  2020-04-03T00:02:19Z

         screen_name
text  \
141   AboubacarKarim  Lorsqu'on a annoncé le confinément, aucun de
n...
172          LesNews  Le maire de #newyork @BilldeBlasio conseille
a...
212           mezmha  Et surtout que l'avenir nous ne le
maîtrisons ...
331           YannBD  Vu qu ils ont sorti 300 milliards pour les
ban...
580        Sarkddine  @Claudy_Siar La réplique sur son compte
@Twitt...
582      nenes_t2020  Voila se que le vaccin pour le #coronavirus
fa...
681     Anonymourist  #CoronavirusFrance  LE TAUX DE MORTALITÉ EN
FR...
720        sbergeron  #COVID19; Vous pensez être admissible à la
Pre...
730        AndyBemba  #Covid19 : 11 nouveaux cas confirmés; 6 de
tra...
749  LeChienDechaine  #Covid_19\nInutile de profiter du
#confinement...
752    MrSmaaashy420  Le #coronavirus va-t-il muter vers une forme
p...
754         AJRVince  Le vaccin Bcg fait parti des vaccins que
nous ...
765     youssef_kaba  Chers agents des FDS, couvre-feu ne signifie
p...
829   OMario07419036  @EPhilippePM .\n#Euthanasie dans les #EHPAD\
n\...
838        sbergeron  #COVID19; Vous pensez être admissible à la
Pre...
903       sputnik_fr  En plein confinement, des Toulousains ont
orga...
935   HuffPostQuebec  Les troubles de l’odorat surviennent
généralem...
965  tahayassinesbai  @UN @ONU_fr @UNHumanRights @UN_Photo
@UNWebTV ...
983     Richie_Hertz  @SpiroAgnewGhost @Eviehome2 Elon is a fraud.
H...
```

```
990    ElleSolitaria  #COVID19\n#COVID19france\n#Confinement\
n#IlsSa...

                  source  reply_to_status_id  reply_to_user_id  \
141   Twitter for iPhone                 NaN               NaN
172   Twitter for iPhone                 NaN               NaN
212  Twitter for Android                 NaN               NaN
331  Twitter for Android                 NaN               NaN
580   Twitter for iPhone        1.245496e+18      3.446271e+08
582   Twitter for iPhone                 NaN               NaN
681      Twitter Web App                 NaN               NaN
720     Twitter for iPad                 NaN               NaN
730   Twitter for iPhone                 NaN               NaN
749      Twitter Web App                 NaN               NaN
752  Twitter for Android                 NaN               NaN
754  Twitter for Android                 NaN               NaN
765  Twitter for Android                 NaN               NaN
829      Twitter Web App        1.245822e+18      1.110890e+09
838     Twitter for iPad                 NaN               NaN
903            TweetDeck                 NaN               NaN
935           SocialFlow                 NaN               NaN
965  Twitter for Android                 NaN      1.415915e+07
983   Twitter for iPhone        1.245863e+18      2.323449e+09
990  Twitter for Android                 NaN               NaN

    reply_to_screen_name  is_quote  ...  country_code  place_full_name
\
141                  NaN     False  ...           NaN              NaN

172                  NaN     False  ...           NaN              NaN

212                  NaN     False  ...           NaN              NaN

331                  NaN     False  ...           NaN              NaN

580          Claudy_Siar     False  ...           NaN              NaN

582                  NaN     False  ...           NaN              NaN

681                  NaN     False  ...           NaN              NaN

720                  NaN     False  ...           NaN              NaN

730                  NaN     False  ...           NaN              NaN

749                  NaN     False  ...           NaN              NaN

752                  NaN     False  ...           NaN              NaN

754                  NaN     False  ...           NaN              NaN
```

```
765                  NaN     False  ...           NaN               NaN
829          EPhilippePM     False  ...           NaN               NaN
838                  NaN     False  ...           NaN               NaN
903                  NaN     False  ...           NaN               NaN
935                  NaN     False  ...           NaN               NaN
965                   UN      True  ...           NaN               NaN
983      SpiroAgnewGhost     False  ...           NaN               NaN
990                  NaN     False  ...           NaN               NaN


     place_type followers_count friends_count account_lang  \
141         NaN              116           131          NaN
172         NaN           398398          1319          NaN
212         NaN              227           373          NaN
331         NaN              775          1462          NaN
580         NaN              203           107          NaN
582         NaN               12            26          NaN
681         NaN               60           635          NaN
720         NaN            13322           633          NaN
730         NaN            36338           840          NaN
749         NaN             8199          7437          NaN
752         NaN             5994          6581          NaN
754         NaN              244           131          NaN
765         NaN              474           287          NaN
829         NaN              186           133          NaN
838         NaN            13322           633          NaN
903         NaN            82229           205          NaN
935         NaN           143092          3712          NaN
965         NaN                4            85          NaN
983         NaN              335           360          NaN
990         NaN                1            22          NaN

       account_created_at  verified  lang lang_count
141  2019-01-07T09:14:24Z     False    fr      30123
172  2009-07-17T19:29:48Z     False    fr      30123
212  2013-08-04T00:30:30Z     False    fr      30123
331  2011-06-08T08:38:37Z     False    fr      30123
580  2012-04-29T09:58:47Z     False    fr      30123
582  2020-03-20T22:53:46Z     False    fr      30123
681  2012-02-22T02:52:34Z     False    fr      30123
720  2008-08-27T17:21:44Z      True    fr      30123
730  2016-11-05T23:31:08Z     False    fr      30123
749  2016-11-13T19:43:58Z     False    fr      30123
```

```
752  2014-08-21T18:36:22Z     False    fr      30123
754  2012-04-22T10:11:08Z     False    fr      30123
765  2011-08-10T17:54:12Z     False    fr      30123
829  2019-07-03T17:45:51Z     False    fr      30123
838  2008-08-27T17:21:44Z      True    fr      30123
903  2011-01-14T09:27:09Z      True    fr      30123
935  2011-10-28T21:33:45Z      True    fr      30123
965  2013-06-30T17:23:47Z     False    fr      30123
983  2019-04-23T17:41:42Z     False    fr      30123
990  2019-12-09T16:50:13Z     False    fr      30123

[20 rows x 23 columns]
```

```
# Looking at the information per column, again like in tweets_df,
there are many missing values in the columns:
# reply_to_status_id, reply_to_user_id, reply_to_screen_name,
country_code, place_full_name, place_type, and account_lang
french_tweets_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 30123 entries, 141 to 536146
Data columns (total 23 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   status_id             30123 non-null  int64
 1   user_id               30123 non-null  int64
 2   created_at            30123 non-null  object
 3   screen_name           30123 non-null  object
 4   text                  30123 non-null  object
 5   source                30123 non-null  object
 6   reply_to_status_id    3395 non-null   float64
 7   reply_to_user_id      3932 non-null   float64
 8   reply_to_screen_name  3932 non-null   object
 9   is_quote              30123 non-null  bool
 10  is_retweet            30123 non-null  bool
 11  favourites_count      30123 non-null  int64
 12  retweet_count         30123 non-null  int64
 13  country_code          945 non-null    object
 14  place_full_name       946 non-null    object
 15  place_type            946 non-null    object
 16  followers_count       30123 non-null  int64
 17  friends_count         30123 non-null  int64
 18  account_lang          0 non-null      float64
 19  account_created_at    30123 non-null  object
 20  verified              30123 non-null  bool
 21  lang                  30123 non-null  object
 22  lang_count            30123 non-null  int64
dtypes: bool(3), float64(3), int64(7), object(10)
memory usage: 4.9+ MB
```

1.5.1.1: Drop Nulls in French Tweets

Similarly to what we did with the English and Spanish tweets, we want to investigate the columns with null values and address them for the French tweets as well. Looking at these, we drop the columns such as reply_to_user_id or place_full_name that have over 75% null values, just like in English and Spanish tweets.

Again, these have an extremely large number of null values, for one, and cannot be easily imputated!

For example, if the country code or place is null, there's no easy imputation to be had. Likewise, whether or not it's a reply is rather meaningless in our analysis, so we can drop it rather than doing null imputation!

```
# Drops columns in french_tweets_df which have 75% of above of their
rows being null
french_tweets_df.dropna(thresh=int(0.75*french_tweets_df.shape[0]),
axis = 1, inplace=True)
french_tweets_df
```

```
/usr/local/lib/python3.8/dist-packages/pandas/util/_decorators.py:311:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return func(*args, **kwargs)
```

	status_id	user_id	created_at
141	1245863595323191296	1082203787962523648	2020-04-03T00:00:02Z
172	1245863602633838603	57732899	2020-04-03T00:00:04Z
212	1245863621332045827	1643982751	2020-04-03T00:00:09Z
331	1245863683260944384	313192351	2020-04-03T00:00:23Z
580	1245863827016519680	566248305	2020-04-03T00:00:58Z
...	...	...	...
536094	1246225920261623808	2533693312	2020-04-03T23:59:47Z
536110	1246225937387204609	1209832080923922432	2020-04-03T23:59:51Z
536112	1246225934220484608	1232524255667027969	2020-04-03T23:59:51Z
536138	1246225958367084545	754094315345969156	2020-04-03T23:59:56Z

```
536146  1246225960111923203   769020181159022593  2020-04-03T23:59:57Z


           screen_name
text  \
141     AboubacarKarim  Lorsqu'on a annoncé le confinément, aucun de
n...
172            LesNews  Le maire de #newyork @BilldeBlasio conseille
a...
212             mezmha  Et surtout que l'avenir nous ne le
maîtrisons ...
331             YannBD  Vu qu ils ont sorti 300 milliards pour les
ban...
580          Sarkddine  @Claudy_Siar La réplique sur son compte
@Twitt...
...                ...
...
536094  lesabotsy_lita  @SafLaBoss Paraît qu’ils sont à l’origine du
#...
536110      SamyHideur  #COVID19, j’étais un visionnaire mdrr
https://...
536112       Dany99325  Aujourd'hui #LCN s'est servi d'une madame
qui ...
536138       carbonewa  Chers compatriotes,\nCette clarification du
do...
536146     Stephie_MKZ  #RDC\n#COVID19 \n\n"Je suis moi-même
congolais...

                     source  is_quote  is_retweet  favourites_count  \
141      Twitter for iPhone     False       False                73
172      Twitter for iPhone     False       False               999
212     Twitter for Android     False       False             10871
331     Twitter for Android     False       False              1110
580      Twitter for iPhone     False       False             23449
...                     ...       ...         ...               ...
536094   Twitter for iPhone     False       False               503
536110   Twitter for iPhone      True       False              1232
536112  Twitter for Android     False       False               732
536138  Twitter for Android      True       False             16412
536146  Twitter for Android     False       False              1216

        retweet_count  followers_count  friends_count
account_created_at  \
141                 8              116            131  2019-01-
07T09:14:24Z
172                20           398398           1319  2009-07-
17T19:29:48Z
212                 1              227            373  2013-08-
04T00:30:30Z
331                 0              775           1462  2011-06-
```

```
08T08:38:37Z
580                 0              203             107  2012-04-
29T09:58:47Z
...               ...              ...             ...
...
536094              0               40             160  2014-05-
07T07:56:29Z
536110              1               18              20  2019-12-
25T13:44:06Z
536112              0               21             117  2020-02-
26T04:35:08Z
536138              1            16182            2501  2016-07-
15T23:24:29Z
536146              3             1274             222  2016-08-
26T03:54:33Z

        verified lang  lang_count
141        False   fr       30123
172        False   fr       30123
212        False   fr       30123
331        False   fr       30123
580        False   fr       30123
...          ...  ...         ...
536094     False   fr       30123
536110     False   fr       30123
536112     False   fr       30123
536138     False   fr       30123
536146     False   fr       30123

[30123 rows x 16 columns]
```

1.5.1.2: Drop Duplicates in French Tweets

We do not want duplicated tweets within our French tweet dataset either. For the same reasons as before, if the same French tweet is used multiple times, then it could skew and alter our overall sentiment that the model would be trained on for the French tweets.

```
# Drops duplicates, keeping the first one to not get rid of all
instances of the tweets and their sentiment
french_tweets_df.drop_duplicates(keep='first', inplace=True)
french_tweets_df
```

```
/usr/local/lib/python3.8/dist-packages/pandas/util/_decorators.py:311:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  return func(*args, **kwargs)
```

```
                  status_id              user_id            created_at
\
141     1245863595323191296  1082203787962523648  2020-04-03T00:00:02Z

172     1245863602633838603            57732899  2020-04-03T00:00:04Z

212     1245863621332045827           1643982751  2020-04-03T00:00:09Z

331     1245863683260944384            313192351  2020-04-03T00:00:23Z

580     1245863827016519680            566248305  2020-04-03T00:00:58Z

...                     ...                  ...                   ...

536094  1246225920261623808           2533693312  2020-04-03T23:59:47Z

536110  1246225937387204609  1209832080923922432  2020-04-03T23:59:51Z

536112  1246225934220484608  1232524255667027969  2020-04-03T23:59:51Z

536138  1246225958367084545   754094315345969156  2020-04-03T23:59:56Z

536146  1246225960111923203   769020181159022593  2020-04-03T23:59:57Z


           screen_name
text  \
141     AboubacarKarim  Lorsqu'on a annoncé le confinément, aucun de
n...
172            LesNews  Le maire de #newyork @BilldeBlasio conseille
a...
212             mezmha  Et surtout que l'avenir nous ne le
maîtrisons ...
331             YannBD  Vu qu ils ont sorti 300 milliards pour les
ban...
580          Sarkddine  @Claudy_Siar La réplique sur son compte
@Twitt...
...                ...
...
536094  lesabotsy_lita  @SafLaBoss Paraît qu’ils sont à l’origine du
#...
536110      SamyHideur  #COVID19, j’étais un visionnaire mdrr
https://...
536112       Dany99325  Aujourd'hui #LCN s'est servi d'une madame
qui ...
536138       carbonewa  Chers compatriotes,\nCette clarification du
do...
536146     Stephie_MKZ  #RDC\n#COVID19 \n\n"Je suis moi-même
congolais...

                    source  is_quote  is_retweet  favourites_count  \
```

```
141      Twitter for iPhone     False       False                73
172      Twitter for iPhone     False       False               999
212     Twitter for Android     False       False             10871
331     Twitter for Android     False       False              1110
580      Twitter for iPhone     False       False             23449
...                     ...       ...         ...               ...
536094   Twitter for iPhone     False       False               503
536110   Twitter for iPhone      True       False              1232
536112  Twitter for Android     False       False               732
536138  Twitter for Android      True       False             16412
536146  Twitter for Android     False       False              1216

        retweet_count  followers_count  friends_count
account_created_at  \
141                 8              116            131  2019-01-
07T09:14:24Z
172                20           398398           1319  2009-07-
17T19:29:48Z
212                 1              227            373  2013-08-
04T00:30:30Z
331                 0              775           1462  2011-06-
08T08:38:37Z
580                 0              203            107  2012-04-
29T09:58:47Z
...               ...              ...            ...
...
536094              0               40            160  2014-05-
07T07:56:29Z
536110              1               18             20  2019-12-
25T13:44:06Z
536112              0               21            117  2020-02-
26T04:35:08Z
536138              1            16182           2501  2016-07-
15T23:24:29Z
536146              3             1274            222  2016-08-
26T03:54:33Z

        verified lang  lang_count
141        False   fr       30123
172        False   fr       30123
212        False   fr       30123
331        False   fr       30123
580        False   fr       30123
...          ...  ...         ...
536094     False   fr       30123
536110     False   fr       30123
536112     False   fr       30123
536138     False   fr       30123
536146     False   fr       30123
```

```
[30123 rows x 16 columns]
```

1.5.1.3: Investigation of French Verified and Followers Count Columns

Just as in the English and Spanish tweet datasets, we want to investigate the discrepancy between how the overall sentiment is for users who are verified versus users who are not verified in the French tweet dataset.

Similarly, we again want to look at the difference between overall sentiment of users with high numbers of followers versus low numbers of followers. We will do cleaning and visualizations of the French verified and followers_count columns to prepare for this avenue of investigation.

Looking at the data, it is interesting to note that the difference between the number of high-follower people and low-follower people is smaller than what the article indicates yet again, with a difference that is comparable to the one exhibited by the English tweets.

This difference could be a reflection of a bias present in the selection of the tweets for the dataset, as it makes sense that users who have high amounts of followers are those who post more and post more about relevant topics (like COVID-19). Thus, this could be an explanation for the trends seen in the comparison bar plots.

```
# Creates a new dataframe of the number of French verified vs non-
verified users
french_verified_df =
french_tweets_df['verified'].value_counts().reset_index()
french_verified_df
```

```
   index  verified
0  False     26818
1   True      3305
```

```
# Inside of verified column, again sets value to 1 if verified and 0
otherwise
french_tweets_df['verified'] =
french_tweets_df['verified'].apply(lambda x: 0 if x == False else 1)
french_tweets_df
```

```
<ipython-input-136-1ae332c23567>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  french_tweets_df['verified'] =
french_tweets_df['verified'].apply(lambda x: 0 if x == False else 1)
```

```
                  status_id              user_id            created_at
\
141     1245863595323191296  1082203787962523648  2020-04-03T00:00:02Z
```

```
172     1245863602633838603              57732899  2020-04-03T00:00:04Z
212     1245863621332045827            1643982751  2020-04-03T00:00:09Z
331     1245863683260944384             313192351  2020-04-03T00:00:23Z
580     1245863827016519680             566248305  2020-04-03T00:00:58Z
...                     ...                   ...                   ...
536094  1246225920261623808            2533693312  2020-04-03T23:59:47Z
536110  1246225937387204609  1209832080923922432  2020-04-03T23:59:51Z
536112  1246225934220484608  1232524255667027969  2020-04-03T23:59:51Z
536138  1246225958367084545   754094315345969156  2020-04-03T23:59:56Z
536146  1246225960111923203   769020181159022593  2020-04-03T23:59:57Z

           screen_name
text  \
141     AboubacarKarim  Lorsqu'on a annoncé le confinément, aucun de
n...
172            LesNews  Le maire de #newyork @BilldeBlasio conseille
a...
212             mezmha  Et surtout que l'avenir nous ne le
maîtrisons ...
331             YannBD  Vu qu ils ont sorti 300 milliards pour les
ban...
580          Sarkddine  @Claudy_Siar La réplique sur son compte
@Twitt...
...                ...
...
536094  lesabotsy_lita  @SafLaBoss Paraît qu’ils sont à l’origine du
#...
536110      SamyHideur  #COVID19, j’étais un visionnaire mdrr
https://...
536112       Dany99325  Aujourd'hui #LCN s'est servi d'une madame
qui ...
536138       carbonewa  Chers compatriotes,\nCette clarification du
do...
536146     Stephie_MKZ  #RDC\n#COVID19 \n\n"Je suis moi-même
congolais...

                    source  is_quote  is_retweet  favourites_count  \
141      Twitter for iPhone     False       False                73
172      Twitter for iPhone     False       False               999
```

```
212     Twitter for Android     False       False             10871
331     Twitter for Android     False       False              1110
580      Twitter for iPhone     False       False             23449
...                     ...       ...         ...               ...
536094   Twitter for iPhone     False       False               503
536110   Twitter for iPhone      True       False              1232
536112  Twitter for Android     False       False               732
536138  Twitter for Android      True       False             16412
536146  Twitter for Android     False       False              1216

        retweet_count  followers_count  friends_count
account_created_at  \
141                 8              116            131  2019-01-
07T09:14:24Z
172                20           398398           1319  2009-07-
17T19:29:48Z
212                 1              227            373  2013-08-
04T00:30:30Z
331                 0              775           1462  2011-06-
08T08:38:37Z
580                 0              203            107  2012-04-
29T09:58:47Z
...               ...              ...            ...
...
536094              0               40            160  2014-05-
07T07:56:29Z
536110              1               18             20  2019-12-
25T13:44:06Z
536112              0               21            117  2020-02-
26T04:35:08Z
536138              1            16182           2501  2016-07-
15T23:24:29Z
536146              3             1274            222  2016-08-
26T03:54:33Z

        verified lang  lang_count
141            0   fr       30123
172            0   fr       30123
212            0   fr       30123
331            0   fr       30123
580            0   fr       30123
...          ...  ...         ...
536094         0   fr       30123
536110         0   fr       30123
536112         0   fr       30123
536138         0   fr       30123
536146         0   fr       30123

[30123 rows x 16 columns]
```

```
# Applies a function to the followers_count column, assigning 1 if
count > 500, representing high followers, and 0 otherwise (low
followers)
french_tweets_df['follower_classification'] =
french_tweets_df['followers_count'].apply(lambda x: 1 if x >= 500 else
0)
french_tweets_df

<ipython-input-137-43b47b45cd6d>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  french_tweets_df['follower_classification'] =
french_tweets_df['followers_count'].apply(lambda x: 1 if x >= 500 else
0)

                  status_id              user_id            created_at
\
141     1245863595323191296  1082203787962523648  2020-04-03T00:00:02Z

172     1245863602633838603             57732899  2020-04-03T00:00:04Z

212     1245863621332045827           1643982751  2020-04-03T00:00:09Z

331     1245863683260944384            313192351  2020-04-03T00:00:23Z

580     1245863827016519680            566248305  2020-04-03T00:00:58Z

...                     ...                  ...                   ...

536094  1246225920261623808           2533693312  2020-04-03T23:59:47Z

536110  1246225937387204609  1209832080923922432  2020-04-03T23:59:51Z

536112  1246225934220484608  1232524255667027969  2020-04-03T23:59:51Z

536138  1246225958367084545   754094315345969156  2020-04-03T23:59:56Z

536146  1246225960111923203   769020181159022593  2020-04-03T23:59:57Z


           screen_name
text  \
141     AboubacarKarim  Lorsqu'on a annoncé le confinément, aucun de
n...
172            LesNews  Le maire de #newyork @BilldeBlasio conseille
a...
212             mezmha  Et surtout que l'avenir nous ne le
```

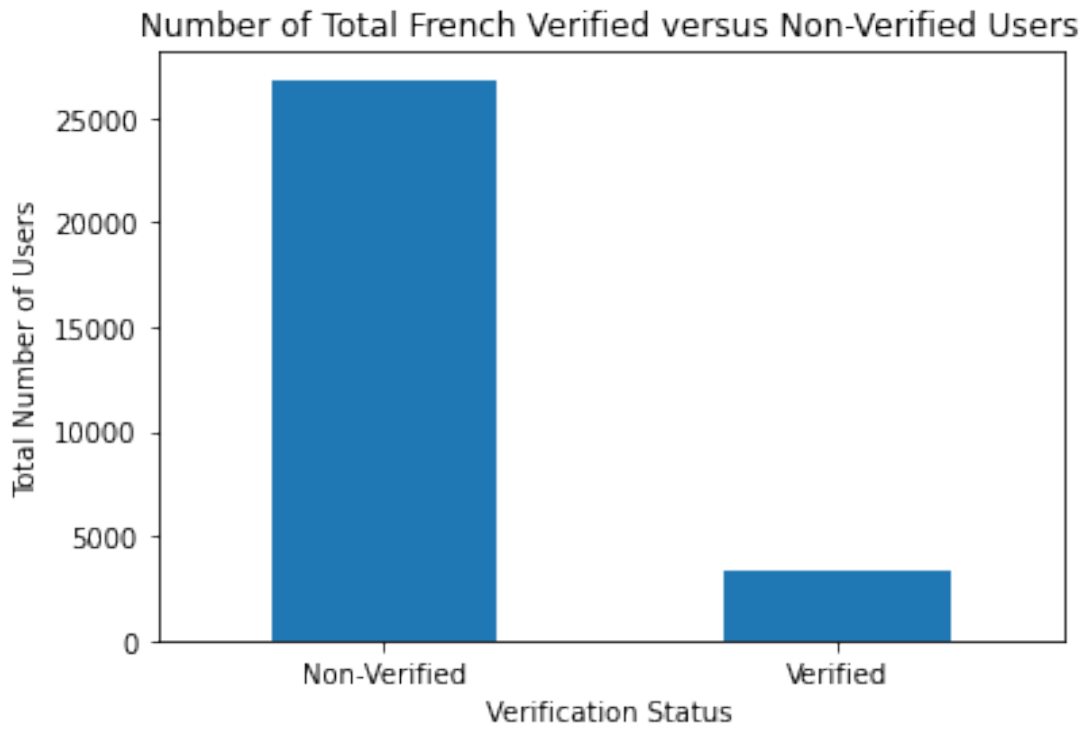
```
maîtrisons ...
331             YannBD  Vu qu ils ont sorti 300 milliards pour les
ban...
580          Sarkddine  @Claudy_Siar La réplique sur son compte
@Twitt...
...                ...
...
536094  lesabotsy_lita  @SafLaBoss Paraît qu’ils sont à l’origine du
#...
536110      SamyHideur  #COVID19, j’étais un visionnaire mdrr
https://...
536112       Dany99325  Aujourd'hui #LCN s'est servi d'une madame
qui ...
536138       carbonewa  Chers compatriotes,\nCette clarification du
do...
536146     Stephie_MKZ  #RDC\n#COVID19 \n\n"Je suis moi-même
congolais...

                     source  is_quote  is_retweet  favourites_count  \
141      Twitter for iPhone     False       False                73
172      Twitter for iPhone     False       False               999
212     Twitter for Android     False       False             10871
331     Twitter for Android     False       False              1110
580      Twitter for iPhone     False       False             23449
...                     ...       ...         ...               ...
536094   Twitter for iPhone     False       False               503
536110   Twitter for iPhone      True       False              1232
536112  Twitter for Android     False       False               732
536138  Twitter for Android      True       False             16412
536146  Twitter for Android     False       False              1216

        retweet_count  followers_count  friends_count
account_created_at  \
141                 8              116            131  2019-01-
07T09:14:24Z
172                20           398398           1319  2009-07-
17T19:29:48Z
212                 1              227            373  2013-08-
04T00:30:30Z
331                 0              775           1462  2011-06-
08T08:38:37Z
580                 0              203            107  2012-04-
29T09:58:47Z
...               ...              ...            ...
...
536094              0               40            160  2014-05-
07T07:56:29Z
536110              1               18             20  2019-12-
25T13:44:06Z
```

```
536112               0               21             117  2020-02-
26T04:35:08Z
536138               1            16182            2501  2016-07-
15T23:24:29Z
536146               3             1274             222  2016-08-
26T03:54:33Z

        verified lang  lang_count  follower_classification
141            0   fr       30123                        0
172            0   fr       30123                        1
212            0   fr       30123                        0
331            0   fr       30123                        1
580            0   fr       30123                        0
...          ...  ...         ...                      ...
536094         0   fr       30123                        0
536110         0   fr       30123                        0
536112         0   fr       30123                        0
536138         0   fr       30123                        1
536146         0   fr       30123                        1

[30123 rows x 17 columns]
```

```
# Creates a new dataframe of the number of high follower vs low
follower users
french_follower_df =
french_tweets_df['follower_classification'].value_counts().reset_index
()
french_follower_df
```

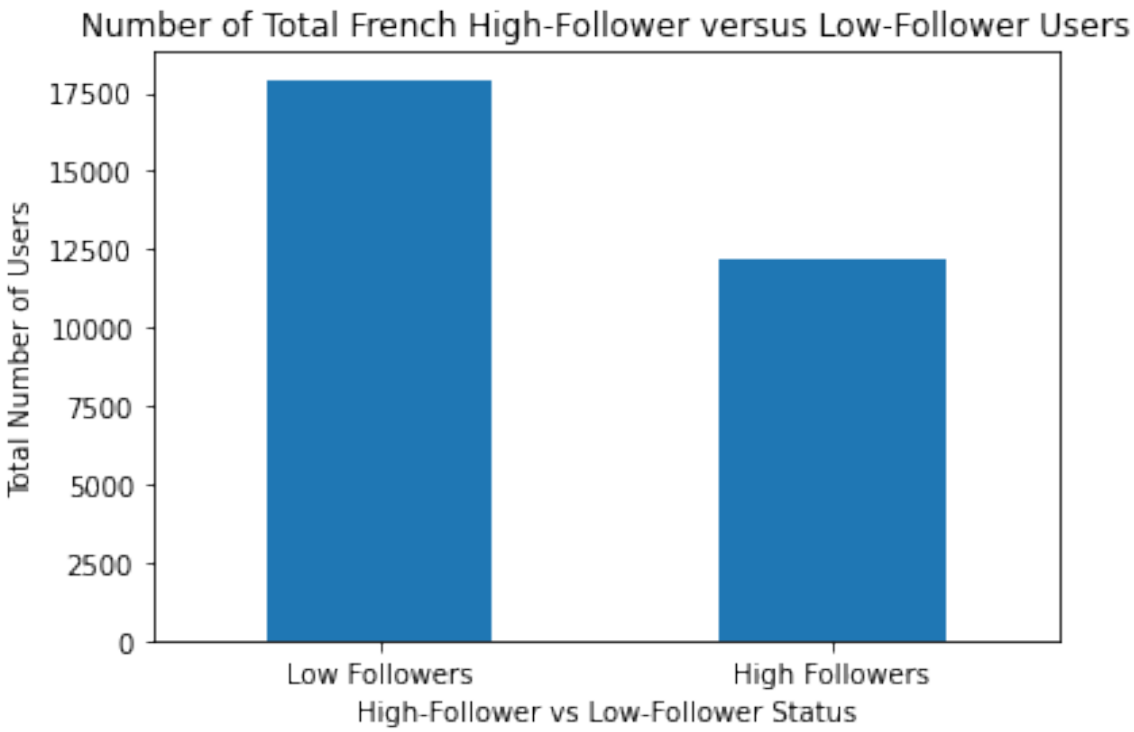
```
   index  follower_classification
0      1                    17917
1      0                    12206
```

```
# Plots number of French verified vs non-verified tweets
french_verified_df.plot.bar(x='index', legend=None)
plt.title("Number of Total French Verified versus Non-Verified Users")
plt.ylabel("Total Number of Users")
plt.xlabel("Verification Status")

bars = ('Non-Verified', 'Verified')
x_pos = np.arange(len(bars))

plt.xticks(x_pos, bars)
plt.xticks(rotation=0)
plt.show()
```



This bar graph shows the large difference between the number of non-verified and verified users -- this makes sense, as naturally there would be less people that are verified, and both classifications still have large sample sizes to try and run analysis on.

```
# Plots number of French high-follower vs low-follower tweets
french_follower_df.plot.bar(x='index', legend=None)
plt.title("Number of Total French High-Follower versus Low-Follower
Users")
plt.ylabel("Total Number of Users")
plt.xlabel("High-Follower vs Low-Follower Status")

bars = ('Low Followers', 'High Followers')
x_pos = np.arange(len(bars))

plt.xticks(x_pos, bars)
plt.xticks(rotation=0)
plt.show()
```



This bar graph shows the difference between the number of high-follower users and low-follower users in the French tweet dataset. This is actually surprising, similar to the English tweet dataset, as given the article it indicated that "high-follower" users were only ~1-2% of the total Twitter userbase. However, the mass majority of Twitter accounts most likely do not post at all, so if a working usable Kaggle dataset was to be compiled, then it is understandable to have an overrepresentation of accounts that post often, and therefore high-follower accounts by association.

1.5.1.4: Cleaning French Tweets Time Stamps

Looking at the French created_at and account_created_at columns, we notice that they need to be converted to proper datetime formatting, as right now they're in an improper format. Thus, we again need to reformat them and transform them to datetime format.

```
# Cleans and converts the created_at and account_created_at columns to
datetime format
french_tweets_df['created_at'] =
french_tweets_df['created_at'].apply(lambda x: x.split('T')[0])
pd.to_datetime(french_tweets_df['created_at'])
french_tweets_df['account_created_at'] =
french_tweets_df['account_created_at'].apply(lambda x: x.split('T')
[0])
pd.to_datetime(french_tweets_df['account_created_at'])
french_tweets_df
```

```
<ipython-input-141-69d6673a1dfc>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  french_tweets_df['created_at'] =
french_tweets_df['created_at'].apply(lambda x: x.split('T')[0])
<ipython-input-141-69d6673a1dfc>:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  french_tweets_df['account_created_at'] =
french_tweets_df['account_created_at'].apply(lambda x: x.split('T')
[0])

                   status_id              user_id  created_at
screen_name  \
141     1245863595323191296  1082203787962523648  2020-04-03
AboubacarKarim
172     1245863602633838603             57732899  2020-04-03
LesNews
212     1245863621332045827           1643982751  2020-04-03
mezmha
331     1245863683260944384            313192351  2020-04-03
YannBD
580     1245863827016519680            566248305  2020-04-03
Sarkddine
...                      ...                  ...         ...
...
536094  1246225920261623808           2533693312  2020-04-03
lesabotsy_lita
536110  1246225937387204609  1209832080923922432  2020-04-03
SamyHideur
536112  1246225934220484608  1232524255667027969  2020-04-03
Dany99325
536138  1246225958367084545   754094315345969156  2020-04-03
carbonewa
536146  1246225960111923203   769020181159022593  2020-04-03
Stephie_MKZ

                                                      text  \
141     Lorsqu'on a annoncé le confinément, aucun de n...
172     Le maire de #newyork @BilldeBlasio conseille a...
212     Et surtout que l'avenir nous ne le maîtrisons ...
331     Vu qu ils ont sorti 300 milliards pour les ban...
580     @Claudy_Siar La réplique sur son compte @Twitt...
...                                                   ...
```

```
536094  @SafLaBoss Paraît qu’ils sont à l’origine du #...
536110  #COVID19, j’étais un visionnaire mdrr https://...
536112  Aujourd'hui #LCN s'est servi d'une madame qui ...
536138  Chers compatriotes,\nCette clarification du do...
536146  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...

                     source  is_quote  is_retweet  favourites_count  \
141      Twitter for iPhone     False       False                73
172      Twitter for iPhone     False       False               999
212     Twitter for Android     False       False             10871
331     Twitter for Android     False       False              1110
580      Twitter for iPhone     False       False             23449
...                     ...       ...         ...               ...
536094   Twitter for iPhone     False       False               503
536110   Twitter for iPhone      True       False              1232
536112  Twitter for Android     False       False               732
536138  Twitter for Android      True       False             16412
536146  Twitter for Android     False       False              1216

        retweet_count  followers_count  friends_count
account_created_at  \
141                 8              116            131          2019-01-
07
172                20           398398           1319          2009-07-
17
212                 1              227            373          2013-08-
04
331                 0              775           1462          2011-06-
08
580                 0              203            107          2012-04-
29
...               ...              ...            ...                .
..
536094              0               40            160          2014-05-
07
536110              1               18             20          2019-12-
25
536112              0               21            117          2020-02-
26
536138              1            16182           2501          2016-07-
15
536146              3             1274            222          2016-08-
26

        verified lang  lang_count  follower_classification
141            0   fr       30123                        0
172            0   fr       30123                        1
212            0   fr       30123                        0
331            0   fr       30123                        1
580            0   fr       30123                        0
```

```
...          ...  ...         ...                        ...
536094         0   fr       30123                          0
536110         0   fr       30123                          0
536112         0   fr       30123                          0
536138         0   fr       30123                          1
536146         0   fr       30123                          1

[30123 rows x 17 columns]
```

1.5.1.5: Investigation of French Tweets Retweets and Quotes and Removing Unnecessary Columns

Again for the French tweets, we only want tweets that are that individual's personal sentiment, rather than quoting another user's opinion. This is because including elements such as retweets and quotes could include multiple users that are retweeting and/or quoting the same tweet, thus again skewing overall sentiment for training.

Thus, as long as they don't account for too many tweets, we should remove all rows that are quotes or tweets.

Likewise, we should remove any columns that are not necessary for overall sentiment analysis, such as personal identifiers.

```
# Checking number of tweets that are retweets
# Seeing that there are 0, we can remove this column easily.
french_tweets_df[french_tweets_df['is_retweet'] == True].count()
```

```
status_id                  0
user_id                    0
created_at                 0
screen_name                0
text                       0
source                     0
is_quote                   0
is_retweet                 0
favourites_count           0
retweet_count              0
followers_count            0
friends_count              0
account_created_at         0
verified                   0
lang                       0
lang_count                 0
follower_classification    0
dtype: int64
```

```
# Checking number of tweets that are quotes.
# Seeing that there's a small proportion of tweets that are quotes,
they can be removed without overly affecting sentiment analysis.
french_tweets_df[french_tweets_df['is_quote'] == True].count()
```

```
status_id                   4260
user_id                     4260
created_at                  4260
screen_name                 4260
text                        4260
source                      4260
is_quote                    4260
is_retweet                  4260
favourites_count            4260
retweet_count               4260
followers_count             4260
friends_count               4260
account_created_at          4260
verified                    4260
lang                        4260
lang_count                  4260
follower_classification     4260
dtype: int64
```

```
# Removes all rows from the dataset that are quotes
french_tweets_df = french_tweets_df[french_tweets_df['is_quote'] ==
False]
french_tweets_df
```

```
                   status_id              user_id  created_at
screen_name  \
141     1245863595323191296  1082203787962523648  2020-04-03
AboubacarKarim
172     1245863602633838603             57732899  2020-04-03
LesNews
212     1245863621332045827           1643982751  2020-04-03
mezmha
331     1245863683260944384            313192351  2020-04-03
YannBD
580     1245863827016519680            566248305  2020-04-03
Sarkddine
...                      ...                  ...         ...
...
535993  1246225828943454208   748886017768390656  2020-04-03
YvesNtsama
536004  1246225841333428225             20956732  2020-04-03
Achigan36
536094  1246225920261623808           2533693312  2020-04-03
lesabotsy_lita
536112  1246225934220484608  1232524255667027969  2020-04-03
Dany99325
536146  1246225960111923203   769020181159022593  2020-04-03
Stephie_MKZ

                                                    text  \
```

```
141     Lorsqu'on a annoncé le confinément, aucun de n...
172     Le maire de #newyork @BilldeBlasio conseille a...
212     Et surtout que l'avenir nous ne le maîtrisons ...
331     Vu qu ils ont sorti 300 milliards pour les ban...
580     @Claudy_Siar La réplique sur son compte @Twitt...
...                                                   ...
535993  @DrManaouda j'ai une question. plusieurs pays ...
536004  Ils ont les masques...on a leur Hydro-électric...
536094  @SafLaBoss Paraît qu’ils sont à l’origine du #...
536112  Aujourd'hui #LCN s'est servi d'une madame qui ...
536146  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...

                     source  is_quote  is_retweet  favourites_count  \
141      Twitter for iPhone     False       False                73
172      Twitter for iPhone     False       False               999
212     Twitter for Android     False       False             10871
331     Twitter for Android     False       False              1110
580      Twitter for iPhone     False       False             23449
...                     ...       ...         ...               ...
535993  Twitter for Android     False       False               190
536004  Twitter for Android     False       False              1326
536094   Twitter for iPhone     False       False               503
536112  Twitter for Android     False       False               732
536146  Twitter for Android     False       False              1216

        retweet_count  followers_count  friends_count
account_created_at  \
141                 8              116            131          2019-01-
07
172                20           398398           1319          2009-07-
17
212                 1              227            373          2013-08-
04
331                 0              775           1462          2011-06-
08
580                 0              203            107          2012-04-
29
...               ...              ...            ...                .
..
535993              0              143            226          2016-07-
01
536004              0               68            143          2009-02-
16
536094              0               40            160          2014-05-
07
536112              0               21            117          2020-02-
26
536146              3             1274            222          2016-08-
26
```

```
        verified lang  lang_count  follower_classification
141            0   fr       30123                        0
172            0   fr       30123                        1
212            0   fr       30123                        0
331            0   fr       30123                        1
580            0   fr       30123                        0
...          ...  ...         ...                      ...
535993         0   fr       30123                        0
536004         0   fr       30123                        0
536094         0   fr       30123                        0
536112         0   fr       30123                        0
536146         0   fr       30123                        1

[25863 rows x 17 columns]
```

```
# Removes all rows from the dataset that are retweets
french_tweets_df = french_tweets_df[french_tweets_df['is_retweet'] ==
False]
french_tweets_df
```

```
                  status_id              user_id  created_at
screen_name  \
141     1245863595323191296  1082203787962523648  2020-04-03
AboubacarKarim
172     1245863602633838603             57732899  2020-04-03
LesNews
212     1245863621332045827           1643982751  2020-04-03
mezmha
331     1245863683260944384            313192351  2020-04-03
YannBD
580     1245863827016519680            566248305  2020-04-03
Sarkddine
...                     ...                  ...         ...
...
535993  1246225828943454208   748886017768390656  2020-04-03
YvesNtsama
536004  1246225841333428225             20956732  2020-04-03
Achigan36
536094  1246225920261623808           2533693312  2020-04-03
lesabotsy_lita
536112  1246225934220484608  1232524255667027969  2020-04-03
Dany99325
536146  1246225960111923203   769020181159022593  2020-04-03
Stephie_MKZ

                                                     text  \
141     Lorsqu'on a annoncé le confinément, aucun de n...
172     Le maire de #newyork @BilldeBlasio conseille a...
212     Et surtout que l'avenir nous ne le maîtrisons ...
331     Vu qu ils ont sorti 300 milliards pour les ban...
```

```
580     @Claudy_Siar La réplique sur son compte @Twitt...
...                                                   ...
535993  @DrManaouda j'ai une question. plusieurs pays ...
536004  Ils ont les masques...on a leur Hydro-électric...
536094  @SafLaBoss Paraît qu’ils sont à l’origine du #...
536112  Aujourd'hui #LCN s'est servi d'une madame qui ...
536146  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...

                     source  is_quote  is_retweet  favourites_count  \
141      Twitter for iPhone     False       False                73
172      Twitter for iPhone     False       False               999
212     Twitter for Android     False       False             10871
331     Twitter for Android     False       False              1110
580      Twitter for iPhone     False       False             23449
...                     ...       ...         ...               ...
535993  Twitter for Android     False       False               190
536004  Twitter for Android     False       False              1326
536094   Twitter for iPhone     False       False               503
536112  Twitter for Android     False       False               732
536146  Twitter for Android     False       False              1216

        retweet_count  followers_count  friends_count
account_created_at  \
141                 8              116            131          2019-01-
07
172                20           398398           1319          2009-07-
17
212                 1              227            373          2013-08-
04
331                 0              775           1462          2011-06-
08
580                 0              203            107          2012-04-
29
...               ...              ...            ...                .
..
535993              0              143            226          2016-07-
01
536004              0               68            143          2009-02-
16
536094              0               40            160          2014-05-
07
536112              0               21            117          2020-02-
26
536146              3             1274            222          2016-08-
26

        verified lang  lang_count  follower_classification
141            0   fr       30123                        0
172            0   fr       30123                        1
212            0   fr       30123                        0
```

```
331            0   fr       30123                          1
580            0   fr       30123                          0
...          ...  ...         ...                        ...
535993         0   fr       30123                          0
536004         0   fr       30123                          0
536094         0   fr       30123                          0
536112         0   fr       30123                          0
536146         0   fr       30123                          1

[25863 rows x 17 columns]
```

```
# As they are now unnecessary, we can remove the is_quote and
is_retweet columns now
french_tweets_df.drop(['is_quote', 'is_retweet'], axis=1,
inplace=True)
french_tweets_df
```

```
                  status_id              user_id  created_at
screen_name  \
141     1245863595323191296  1082203787962523648  2020-04-03
AboubacarKarim
172     1245863602633838603             57732899  2020-04-03
LesNews
212     1245863621332045827           1643982751  2020-04-03
mezmha
331     1245863683260944384            313192351  2020-04-03
YannBD
580     1245863827016519680            566248305  2020-04-03
Sarkddine
...                     ...                  ...         ...
...
535993  1246225828943454208   748886017768390656  2020-04-03
YvesNtsama
536004  1246225841333428225            20956732  2020-04-03
Achigan36
536094  1246225920261623808           2533693312  2020-04-03
lesabotsy_lita
536112  1246225934220484608  1232524255667027969  2020-04-03
Dany99325
536146  1246225960111923203   769020181159022593  2020-04-03
Stephie_MKZ

                                                    text  \
141     Lorsqu'on a annoncé le confinément, aucun de n...
172     Le maire de #newyork @BilldeBlasio conseille a...
212     Et surtout que l'avenir nous ne le maîtrisons ...
331     Vu qu ils ont sorti 300 milliards pour les ban...
580     @Claudy_Siar La réplique sur son compte @Twitt...
...                                                   ...
535993  @DrManaouda j'ai une question. plusieurs pays ...
```

```
536004  Ils ont les masques...on a leur Hydro-électric...
536094  @SafLaBoss Paraît qu’ils sont à l’origine du #...
536112  Aujourd'hui #LCN s'est servi d'une madame qui ...
536146  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...

                     source  favourites_count  retweet_count
followers_count  \
141      Twitter for iPhone               73              8
116
172      Twitter for iPhone              999             20
398398
212     Twitter for Android             10871              1
227
331     Twitter for Android              1110              0
775
580      Twitter for iPhone             23449              0
203
...                     ...               ...            ...
...
535993  Twitter for Android               190              0
143
536004  Twitter for Android              1326              0
68
536094   Twitter for iPhone               503              0
40
536112  Twitter for Android               732              0
21
536146  Twitter for Android              1216              3
1274

        friends_count account_created_at  verified lang  lang_count  \
141               131         2019-01-07         0   fr       30123
172              1319         2009-07-17         0   fr       30123
212               373         2013-08-04         0   fr       30123
331              1462         2011-06-08         0   fr       30123
580               107         2012-04-29         0   fr       30123
...               ...                ...       ...  ...         ...
535993            226         2016-07-01         0   fr       30123
536004            143         2009-02-16         0   fr       30123
536094            160         2014-05-07         0   fr       30123
536112            117         2020-02-26         0   fr       30123
536146            222         2016-08-26         0   fr       30123

        follower_classification
141                           0
172                           1
212                           0
331                           1
580                           0
...                         ...
```

```
535993                          0
536004                          0
536094                          0
536112                          0
536146                          1

[25863 rows x 15 columns]
```

```
# Looking at the status_id, user_id, and screen_name columns, we don't
need them for French tweet sentiment analysis, so we can drop them.
french_tweets_df.drop(['status_id', 'user_id', 'screen_name'], axis=1,
inplace=True)
french_tweets_df
```

```
        created_at                                                  text
\
141     2020-04-03  Lorsqu'on a annoncé le confinément, aucun de n...

172     2020-04-03  Le maire de #newyork @BilldeBlasio conseille a...

212     2020-04-03  Et surtout que l'avenir nous ne le maîtrisons ...

331     2020-04-03  Vu qu ils ont sorti 300 milliards pour les ban...

580     2020-04-03  @Claudy_Siar La réplique sur son compte @Twitt...

...            ...                                                 ...

535993  2020-04-03  @DrManaouda j'ai une question. plusieurs pays ...

536004  2020-04-03  Ils ont les masques...on a leur Hydro-électric...

536094  2020-04-03  @SafLaBoss Paraît qu’ils sont à l’origine du #...

536112  2020-04-03  Aujourd'hui #LCN s'est servi d'une madame qui ...

536146  2020-04-03  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...


                      source  favourites_count  retweet_count
followers_count  \
141      Twitter for iPhone                73              8
116
172      Twitter for iPhone               999             20
398398
212     Twitter for Android             10871              1
227
331     Twitter for Android              1110              0
775
580      Twitter for iPhone             23449              0
203
```

```
...                     ...                ...             ...
...
535993  Twitter for Android                190               0
143
536004  Twitter for Android               1326               0
68
536094   Twitter for iPhone                503               0
40
536112  Twitter for Android                732               0
21
536146  Twitter for Android               1216               3
1274

        friends_count account_created_at  verified lang  lang_count  \
141               131         2019-01-07         0   fr       30123
172              1319         2009-07-17         0   fr       30123
212               373         2013-08-04         0   fr       30123
331              1462         2011-06-08         0   fr       30123
580               107         2012-04-29         0   fr       30123
...               ...                ...       ...  ...         ...
535993            226         2016-07-01         0   fr       30123
536004            143         2009-02-16         0   fr       30123
536094            160         2014-05-07         0   fr       30123
536112            117         2020-02-26         0   fr       30123
536146            222         2016-08-26         0   fr       30123

        follower_classification
141                           0
172                           1
212                           0
331                           1
580                           0
...                         ...
535993                        0
536004                        0
536094                        0
536112                        0
536146                        1

[25863 rows x 12 columns]
```

```
# Furthermore, we also do not need the source column for our analysis,
as evidenced through analysis in Spanish and English tweets
# Thus, we can remove this.
# Likewise, we no longer need the language columns for our analysis,
so lang_count and lang (redundant) can be removed too
french_tweets_df.drop(['source', 'lang', 'lang_count'], axis=1,
inplace=True)
french_tweets_df
```

```
        created_at                                                  text
\
141     2020-04-03  Lorsqu'on a annoncé le confinément, aucun de n...

172     2020-04-03  Le maire de #newyork @BilldeBlasio conseille a...

212     2020-04-03  Et surtout que l'avenir nous ne le maîtrisons ...

331     2020-04-03  Vu qu ils ont sorti 300 milliards pour les ban...

580     2020-04-03  @Claudy_Siar La réplique sur son compte @Twitt...

...            ...                                                 ...

535993  2020-04-03  @DrManaouda j'ai une question. plusieurs pays ...

536004  2020-04-03  Ils ont les masques...on a leur Hydro-électric...

536094  2020-04-03  @SafLaBoss Paraît qu’ils sont à l’origine du #...

536112  2020-04-03  Aujourd'hui #LCN s'est servi d'une madame qui ...

536146  2020-04-03  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...


        favourites_count  retweet_count  followers_count
friends_count  \
141                   73              8              116
131
172                  999             20           398398
1319
212                10871              1              227
373
331                 1110              0              775
1462
580                23449              0              203
107
...                  ...            ...              ...              ..
.
535993               190              0              143
226
536004              1326              0               68
143
536094               503              0               40
160
536112               732              0               21
117
536146              1216              3             1274
222

       account_created_at  verified  follower_classification
```

```
141             2019-01-07         0                          0
172             2009-07-17         0                          1
212             2013-08-04         0                          0
331             2011-06-08         0                          1
580             2012-04-29         0                          0
...                    ...       ...                        ...
535993          2016-07-01         0                          0
536004          2009-02-16         0                          0
536094          2014-05-07         0                          0
536112          2020-02-26         0                          0
536146          2016-08-26         0                          1

[25863 rows x 9 columns]
```

1.5.2 French Tweets Sentiment Analysis

1.5.2.1: French Sentiment Analysis Preprocessing

Uses the NLTK library, specifically the French parts of it.

We have to again first tokenize each tweet, before turning them into lowercase, stemming them with the SnowballStemmer library, and removing any non-alphabetic characters or "stopwords" according to the French library.

Because the SnowballStemmer library is an improvement over the normal PorterStemmer library, and because it is included in the NLTK library and supports multi-lingual analysis for languages like French as well, we have chosen to use that to increase the accuracy of our sentiment analysis for French tweets.

```
# Install the French stopwords set
from nltk.corpus import stopwords
nltk.download('stopwords')
french_stopwords = set(stopwords.words('french'))
```

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

```
# Install the French SnowballStemmer from the NLTK Snowball Stemmer
library
from nltk.stem.snowball import SnowballStemmer
french_snowball_stemmer = SnowballStemmer(language='french')

# A user-defined function that tokenizes, lowercases, stems, and
removes stopwords from the tweet and returns the tokenized form
@numba.jit
def tokenize_french_content(content):
  tokens = nltk.word_tokenize(content, language='french')
  final_string = []
  for tk in tokens:
    lowercase = tk.lower()
```

```
    french_stemmed_lowercase = french_snowball_stemmer.stem(lowercase)
    if ((str.isalpha(french_stemmed_lowercase)) and
(french_stemmed_lowercase not in french_stopwords)):
      final_string.append(french_stemmed_lowercase)
    else:
      continue

  return final_string

# Applies the function on each tweet in the text column
french_tweets_df['processed_text'] =
french_tweets_df['text'].map(lambda x: tokenize_french_content(x))
french_tweets_df
```

```
<ipython-input-151-1691ee4b8ee1>:2: NumbaWarning:
Compilation is falling back to object mode WITH looplifting enabled
because Function "tokenize_french_content" failed type inference due
to: Untyped global name 'french_snowball_stemmer': Cannot determine
Numba type of <class 'nltk.stem.snowball.SnowballStemmer'>

File "<ipython-input-151-1691ee4b8ee1>", line 8:
def tokenize_french_content(content):
    <source elided>
    lowercase = tk.lower()
    french_stemmed_lowercase = french_snowball_stemmer.stem(lowercase)
    ^

  @numba.jit
<ipython-input-151-1691ee4b8ee1>:2: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "tokenize_french_content" failed type inference due
to: Cannot determine Numba type of <class
'numba.core.dispatcher.LiftedLoop'>

File "<ipython-input-151-1691ee4b8ee1>", line 6:
def tokenize_french_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "tokenize_french_content" was compiled
in object mode without forceobj=True, but has lifted loops.

File "<ipython-input-151-1691ee4b8ee1>", line 4:
def tokenize_french_content(content):
  tokens = nltk.word_tokenize(content, language='french')
  ^
```

```
  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-151-1691ee4b8ee1>", line 4:
def tokenize_french_content(content):
  tokens = nltk.word_tokenize(content, language='french')
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,
<ipython-input-151-1691ee4b8ee1>:2: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "tokenize_french_content" failed type inference due
to: Untyped global name 'french_snowball_stemmer': Cannot determine
Numba type of <class 'nltk.stem.snowball.SnowballStemmer'>

File "<ipython-input-151-1691ee4b8ee1>", line 8:
def tokenize_french_content(content):
    <source elided>
    lowercase = tk.lower()
    french_stemmed_lowercase = french_snowball_stemmer.stem(lowercase)
    ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "tokenize_french_content" was compiled
in object mode without forceobj=True.

File "<ipython-input-151-1691ee4b8ee1>", line 6:
def tokenize_french_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
```

```
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-151-1691ee4b8ee1>", line 6:
def tokenize_french_content(content):
    <source elided>
  final_string = []
  for tk in tokens:
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,

        created_at                                                text
\
141     2020-04-03  Lorsqu'on a annoncé le confinément, aucun de n...

172     2020-04-03  Le maire de #newyork @BilldeBlasio conseille a...

212     2020-04-03  Et surtout que l'avenir nous ne le maîtrisons ...

331     2020-04-03  Vu qu ils ont sorti 300 milliards pour les ban...

580     2020-04-03  @Claudy_Siar La réplique sur son compte @Twitt...

...            ...                                                ...

535993  2020-04-03  @DrManaouda j'ai une question. plusieurs pays ...

536004  2020-04-03  Ils ont les masques...on a leur Hydro-électric...

536094  2020-04-03  @SafLaBoss Paraît qu’ils sont à l’origine du #...

536112  2020-04-03  Aujourd'hui #LCN s'est servi d'une madame qui ...

536146  2020-04-03  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...


        favourites_count  retweet_count  followers_count
friends_count  \
141                   73              8              116
131
172                  999             20           398398
1319
212                10871              1              227
373
331                 1110              0              775
1462
580                23449              0              203
107
...                  ...            ...              ...             ..
.
535993               190              0              143
```

```
226
536004                1326                 0                  68
143
536094                 503                 0                  40
160
536112                 732                 0                  21
117
536146                1216                 3                1274
222

       account_created_at  verified  follower_classification  \
141            2019-01-07         0                         0
172            2009-07-17         0                         1
212            2013-08-04         0                         0
331            2011-06-08         0                         1
580            2012-04-29         0                         0
...                   ...       ...                       ...
535993         2016-07-01         0                         0
536004         2009-02-16         0                         0
536094         2014-05-07         0                         0
536112         2020-02-26         0                         0
536146         2016-08-26         0                         1

                                              processed_text
141     [a, annonc, confin, aucun, précip, achet, terr...
172     [mair, newyork, billdeblasio, conseil, port, m...
212                [surtout, maîtrison, coronavirus, http]
331     [vu, sort, milliard, banqu, entrepris, a, mill...
580     [répliqu, compt, twitt, encor, plus, insult, e...
...                                                   ...
535993  [drmanaoud, question, plusieur, pay, europ, pr...
536004  [masqu, a, pens, avant, export, ver, canad, po...
536094                          [saflaboss, paraît, origin]
536112  [lcn, serv, madam, souffr, rhumat, fair, croir...
536146  [rdc, congol, peux, utilis, congol, comm, coba...

[25863 rows x 10 columns]
```

```
# A full similar user-defined function to return a cleaned, but not
split into tokens version of the tweet for model training purposes
@numba.jit
def clean_tweet_french(content):
  final_string = ""
  tokens = nltk.word_tokenize(content)
  for word in tokens:
    stemmed = word.lower()
    if(stemmed not in french_stopwords):
      for char in range(0, len(stemmed)):
        stemmed = stemmed.replace('#', '')
        stemmed = stemmed.replace('@', '')
```

```
      final_string = final_string + stemmed + " "
    else:
      continue
  return final_string

# Applying that user-defined function to the text column to create a
new training_text column
french_tweets_df['training_text'] =
french_tweets_df['text'].apply(lambda x: clean_tweet_french(x))
french_tweets_df
```

```
<ipython-input-153-8c15da255171>:2: NumbaWarning:
Compilation is falling back to object mode WITH looplifting enabled
because Function "clean_tweet_french" failed type inference due to:
Unknown attribute 'word_tokenize' of type Module(<module 'nltk' from
'/usr/local/lib/python3.8/dist-packages/nltk/__init__.py'>)

File "<ipython-input-153-8c15da255171>", line 5:
def clean_tweet_french(content):
    <source elided>
  final_string = ""
  tokens = nltk.word_tokenize(content)
  ^

During: typing of get attribute at <ipython-input-153-8c15da255171>
(5)

File "<ipython-input-153-8c15da255171>", line 5:
def clean_tweet_french(content):
    <source elided>
  final_string = ""
  tokens = nltk.word_tokenize(content)
  ^

  @numba.jit
<ipython-input-153-8c15da255171>:2: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled
because Function "clean_tweet_french" failed type inference due to:
Cannot determine Numba type of <class
'numba.core.dispatcher.LiftedLoop'>

File "<ipython-input-153-8c15da255171>", line 6:
def clean_tweet_french(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  @numba.jit
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
```

```
y:151: NumbaWarning: Function "clean_tweet_french" was compiled in
object mode without forceobj=True, but has lifted loops.

File "<ipython-input-153-8c15da255171>", line 4:
def clean_tweet_french(content):
  final_string = ""
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-153-8c15da255171>", line 4:
def clean_tweet_french(content):
  final_string = ""
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:151: NumbaWarning: Function "clean_tweet_french" was compiled in
object mode without forceobj=True.

File "<ipython-input-153-8c15da255171>", line 6:
def clean_tweet_french(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
  ^

  warnings.warn(errors.NumbaWarning(warn_msg,
/usr/local/lib/python3.8/dist-packages/numba/core/object_mode_passes.p
y:161: NumbaDeprecationWarning:
Fall-back from the nopython compilation path to the object mode
compilation path has been detected, this is deprecated behaviour.

For more information visit
https://numba.readthedocs.io/en/stable/reference/deprecation.html#depr
ecation-of-object-mode-fall-back-behaviour-when-using-jit

File "<ipython-input-153-8c15da255171>", line 6:
def clean_tweet_french(content):
    <source elided>
  tokens = nltk.word_tokenize(content)
  for word in tokens:
```

```
  ^

  warnings.warn(errors.NumbaDeprecationWarning(msg,

        created_at                                                text
\
141     2020-04-03  Lorsqu'on a annoncé le confinément, aucun de n...

172     2020-04-03  Le maire de #newyork @BilldeBlasio conseille a...

212     2020-04-03  Et surtout que l'avenir nous ne le maîtrisons ...

331     2020-04-03  Vu qu ils ont sorti 300 milliards pour les ban...

580     2020-04-03  @Claudy_Siar La réplique sur son compte @Twitt...

...            ...                                                ...

535993  2020-04-03  @DrManaouda j'ai une question. plusieurs pays ...

536004  2020-04-03  Ils ont les masques...on a leur Hydro-électric...

536094  2020-04-03  @SafLaBoss Paraît qu’ils sont à l’origine du #...

536112  2020-04-03  Aujourd'hui #LCN s'est servi d'une madame qui ...

536146  2020-04-03  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...


        favourites_count  retweet_count  followers_count
friends_count  \
141                   73              8              116
131
172                  999             20           398398
1319
212                10871              1              227
373
331                 1110              0              775
1462
580                23449              0              203
107
...                  ...            ...              ...             ..
.
535993               190              0              143
226
536004              1326              0               68
143
536094               503              0               40
160
536112               732              0               21
117
```

```
536146              1216               3             1274
222

       account_created_at  verified  follower_classification  \
141            2019-01-07         0                        0
172            2009-07-17         0                        1
212            2013-08-04         0                        0
331            2011-06-08         0                        1
580            2012-04-29         0                        0
...                   ...       ...                      ...
535993         2016-07-01         0                        0
536004         2009-02-16         0                        0
536094         2014-05-07         0                        0
536112         2020-02-26         0                        0
536146         2016-08-26         0                        1

                                           processed_text  \
141     [a, annonc, confin, aucun, précip, achet, terr...
172     [mair, newyork, billdeblasio, conseil, port, m...
212               [surtout, maîtrison, coronavirus, http]
331     [vu, sort, milliard, banqu, entrepris, a, mill...
580     [répliqu, compt, twitt, encor, plus, insult, e...
...                                                   ...
535993  [drmanaoud, question, plusieur, pay, europ, pr...
536004  [masqu, a, pens, avant, export, ver, canad, po...
536094                      [saflaboss, paraît, origin]
536112  [lcn, serv, madam, souffr, rhumat, fair, croir...
536146  [rdc, congol, peux, utilis, congol, comm, coba...

                                            training_text
141     lorsqu'on a annoncé confinément , aucun s'est ...
172     maire  newyork  billdeblasio conseille new-yor...
212     surtout l'avenir maîtrisons .  coronavirus htt...
331     vu sorti 300 milliards banques entreprises , a...
580      claudy_siar réplique compte  twitter encore p...
...                                                   ...
535993   drmanaouda j'ai question . plusieurs pays eur...
536004  masques ... a hydro-électricité ... ' penserai...
536094            saflaboss paraît ’ ’ origine  covid19
536112  aujourd'hui  lcn s'est servi d'une madame souf...
536146   rdc  covid19 '' moi-même congolais , peux uti...

[25863 rows x 11 columns]
```

1.5.2.2: French Sentiment Analysis Calculation Using vaderSentiment-fr

For our French tweet sentiment analysis calculation, we will be using the vaderSentiment-fr library. (https://github.com/thomas7lieues/vader_FR)

vaderSentiment-fr is a sentiment analysis and natural language processing library based on the English vaderSentiment library that we used earlier to analyze the English tweets. It is trained on French-language social media entries, and more that has been used to study sentiment in multiple academic studies.

For example, it has been used to investigate news articles talking about food security (https://link.springer.com/chapter/10.1007/978-3-031-16564-1_7). Admittedly there have been less research studies using this library, but as it is based on the widely-used English vaderSentiment library, we considered it suitable.

Thus, we felt that for the purposes of our analysis, which is to investigate multilingual sentiment in tweets regarding COVID-19, that the vaderSentiment-fr library would be a good fit. Because of its usage by multiple studies in reputable journals, and its basis being the widely-used English VaderSentiment library that we used earlier, this toolkit was a good fit for what we wished to do.

```
# Install the vaderSentiment-fr library
!pip install vaderSentiment.fr
```

```
Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Collecting vaderSentiment.fr
  Downloading vaderSentiment_fr-1.3.4.tar.gz (187 kB)
ent.fr
  Building wheel for vaderSentiment.fr (setup.py) ... ent.fr:
filename=vaderSentiment_fr-1.3.4-py3-none-any.whl size=185986
sha256=d17421a2e894e9be17b0561821f901d20a5e107c915c6b9d508a3d1aa379457
f
  Stored in directory:
/root/.cache/pip/wheels/a6/a6/8b/6f8461bbebec0a38f5068fc8331472b4a2ae7
d614bb3ec2f71
Successfully built vaderSentiment.fr
Installing collected packages: unidecode, fuzzywuzzy,
vaderSentiment.fr
Successfully installed fuzzywuzzy-0.18.0 unidecode-1.3.6
vaderSentiment.fr-1.3.4
```

```
# Get SentimentIntensityAnalyzer from vaderSentiment-fr
from vaderSentiment_fr.vaderSentiment import
SentimentIntensityAnalyzer
```

```
/usr/local/lib/python3.8/dist-packages/fuzzywuzzy/fuzz.py:11:
UserWarning: Using slow pure-python SequenceMatcher. Install python-
Levenshtein to remove this warning
  warnings.warn('Using slow pure-python SequenceMatcher. Install
python-Levenshtein to remove this warning')
```

```
# Creates a SentimentIntensityAnalyzer object
french_sia = SentimentIntensityAnalyzer()

# Creates a user-defined function to get compound sentiment per
sentence
```

```
def retrieve_french_sentiment(content):
  sentence = ' '.join(word for word in content)
  return french_sia.polarity_scores(sentence)['compound']

# Applies the function to each processed text to get sentiment
french_tweets_df['sentiment'] =
french_tweets_df['processed_text'].apply(lambda x:
retrieve_french_sentiment(x))
french_tweets_df
```

```
        created_at                                                text
\
141     2020-04-03  Lorsqu'on a annoncé le confinément, aucun de n...

172     2020-04-03  Le maire de #newyork @BilldeBlasio conseille a...

212     2020-04-03  Et surtout que l'avenir nous ne le maîtrisons ...

331     2020-04-03  Vu qu ils ont sorti 300 milliards pour les ban...

580     2020-04-03  @Claudy_Siar La réplique sur son compte @Twitt...

...            ...                                                ...

535993  2020-04-03  @DrManaouda j'ai une question. plusieurs pays ...

536004  2020-04-03  Ils ont les masques...on a leur Hydro-électric...

536094  2020-04-03  @SafLaBoss Paraît qu’ils sont à l’origine du #...

536112  2020-04-03  Aujourd'hui #LCN s'est servi d'une madame qui ...

536146  2020-04-03  #RDC\n#COVID19 \n\n"Je suis moi-même congolais...


        favourites_count  retweet_count  followers_count
friends_count  \
141                   73              8              116
131
172                  999             20           398398
1319
212                10871              1              227
373
331                 1110              0              775
1462
580                23449              0              203
107
...                  ...            ...              ...             ..
.
535993               190              0              143
226
```

```
536004               1326               0                68
143
536094                503               0                40
160
536112                732               0                21
117
536146               1216               3              1274
222

       account_created_at  verified  follower_classification  \
141            2019-01-07         0                        0
172            2009-07-17         0                        1
212            2013-08-04         0                        0
331            2011-06-08         0                        1
580            2012-04-29         0                        0
...                   ...       ...                      ...
535993         2016-07-01         0                        0
536004         2009-02-16         0                        0
536094         2014-05-07         0                        0
536112         2020-02-26         0                        0
536146         2016-08-26         0                        1

                                          processed_text  \
141     [a, annonc, confin, aucun, précip, achet, terr...
172     [mair, newyork, billdeblasio, conseil, port, m...
212               [surtout, maîtrison, coronavirus, http]
331     [vu, sort, milliard, banqu, entrepris, a, mill...
580     [répliqu, compt, twitt, encor, plus, insult, e...
...                                                   ...
535993  [drmanaoud, question, plusieur, pay, europ, pr...
536004  [masqu, a, pens, avant, export, ver, canad, po...
536094                        [saflaboss, paraît, origin]
536112  [lcn, serv, madam, souffr, rhumat, fair, croir...
536146  [rdc, congol, peux, utilis, congol, comm, coba...

                                         training_text  sentiment
141     lorsqu'on a annoncé confinément , aucun s'est ...     0.0000
172     maire  newyork  billdeblasio conseille new-yor...     0.0000
212     surtout l'avenir maîtrisons .  coronavirus htt...     0.0000
331     vu sorti 300 milliards banques entreprises , a...     0.0000
580      claudy_siar réplique compte  twitter encore p...     0.4767
...                                                   ...        ...
535993   drmanaouda j'ai question . plusieurs pays eur...     0.6597
536004  masques ... a hydro-électricité ... ' penserai...     0.0000
536094            saflaboss paraît ’ ’ origine  covid19      0.0000
536112  aujourd'hui  lcn s'est servi d'une madame souf...     0.0000
536146   rdc  covid19 '' moi-même congolais , peux uti...     0.0000

[25863 rows x 12 columns]
```

```
# Looks at the numerical vs categorical columns for later separation
french_tweets_df.dtypes
```

```
created_at                  object
text                        object
favourites_count             int64
retweet_count                int64
followers_count              int64
friends_count                int64
account_created_at          object
verified                     int64
follower_classification      int64
processed_text              object
training_text               object
sentiment                  float64
dtype: object
```

```
# Creates a Dataframe out of the numerical columns for
multicolinearity investigaiton
french_numerics_df = french_tweets_df[['favourites_count',
'retweet_count', 'verified',
                                       'follower_classification',
'friends_count', 'sentiment']]
french_numerics_df
```

```
        favourites_count  retweet_count  verified
follower_classification  \
141                   73              8         0
0
172                  999             20         0
1
212                10871              1         0
0
331                 1110              0         0
1
580                23449              0         0
0
...                  ...            ...       ...
...
535993               190              0         0
0
536004              1326              0         0
0
536094               503              0         0
0
536112               732              0         0
0
536146              1216              3         0
1
```

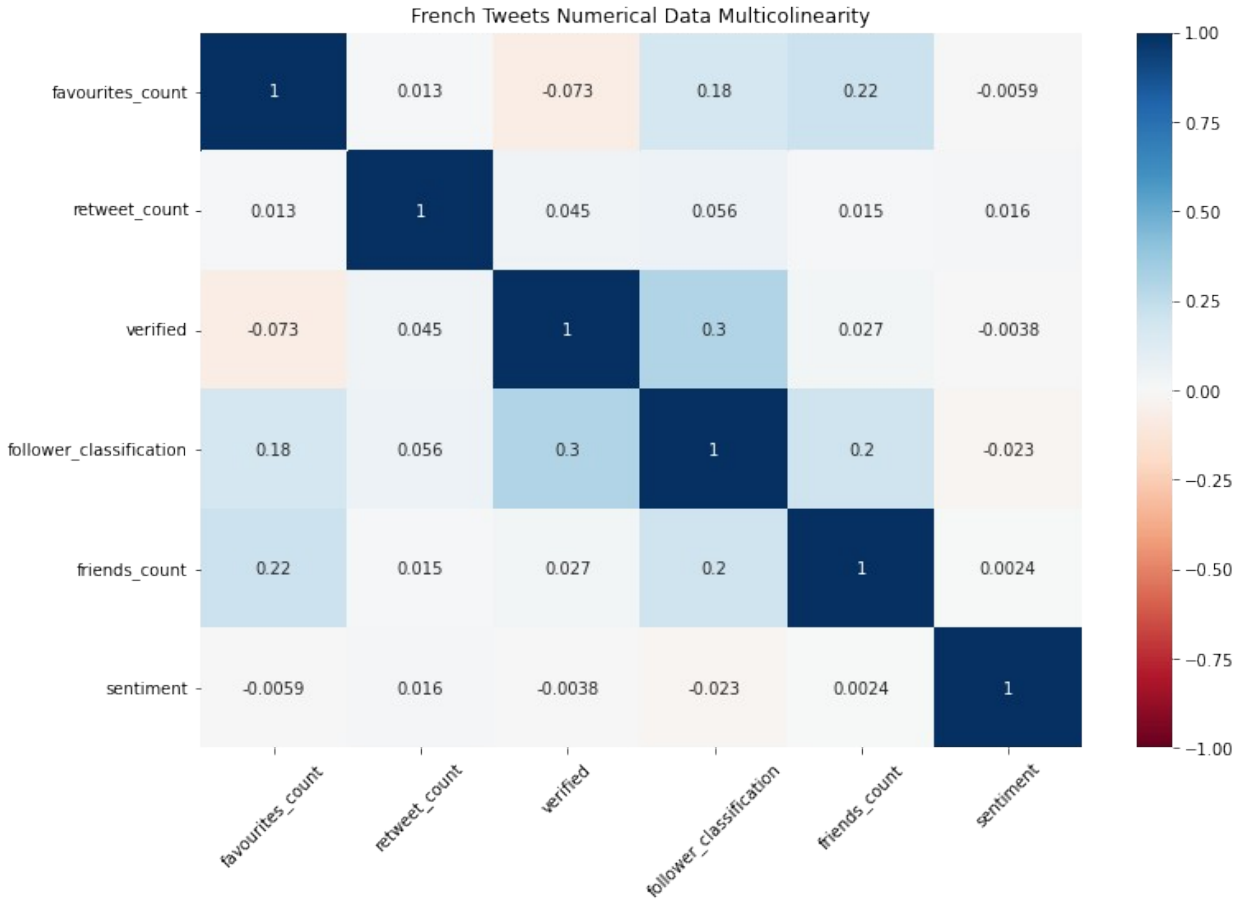
```
        friends_count  sentiment
141               131     0.0000
172              1319     0.0000
212               373     0.0000
331              1462     0.0000
580               107     0.4767
...               ...        ...
535993            226     0.6597
536004            143     0.0000
536094            160     0.0000
536112            117     0.0000
536146            222     0.0000

[25863 rows x 6 columns]
```

1.5.3 French Tweets Visualizations

As with English and Spanish, we will create a heatmap to visualize multicollinearity and KDE plots to visualize the distribution of sentiment across different demographics.

```
# Correlation matrix of a heatmap of the numerical columns for the
French tweets
plt.figure(figsize=(12, 8))
french_corr_matrix = sns.heatmap(french_numerics_df.corr(), vmin=-1,
vmax=1, cmap='RdBu', annot=True)
plt.title('French Tweets Numerical Data Multicolinearity')
plt.xticks(rotation=45)
```

```
(array([0.5, 1.5, 2.5, 3.5, 4.5, 5.5]),
 <a list of 6 Text major ticklabel objects>)
```



The results of this correlation heatmap show that none of the variables are too colinear, so all of them can be used for future analysis and modeling.

We will create a word cloud as before.

```
# Create the top tokens from our tokenized text. Due to the way our
tokenizer processed French,
# "https" is treated as a token and became the most popular token. To
offset this, we only count
# tokens if they're not equal to "https"
top_tokens_list_french = french_tweets_df['processed_text']
top_tokens_french = []
for sublist in top_tokens_list_french:
    for element in sublist:
      if(element != 'http'):
        top_tokens_french.append(element)

#Count each token and collect the top 20 most frequent ones
from collections import Counter
cnt = Counter()
for word in top_tokens_french:
```

```
    cnt[word] += 1
top_most_common_french = cnt.most_common(20)

#Plot the word cloud
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
w = WordCloud(background_color='white')

cnt2 = Counter()
for word in top_tokens_french:
    cnt2[word] += 1

plt.figure(figsize=(12, 8))
w.generate_from_frequencies(cnt2)
plt.title('Wordcloud of Most Frequently Seen Words in French Tweets')
plt.imshow(w)
```

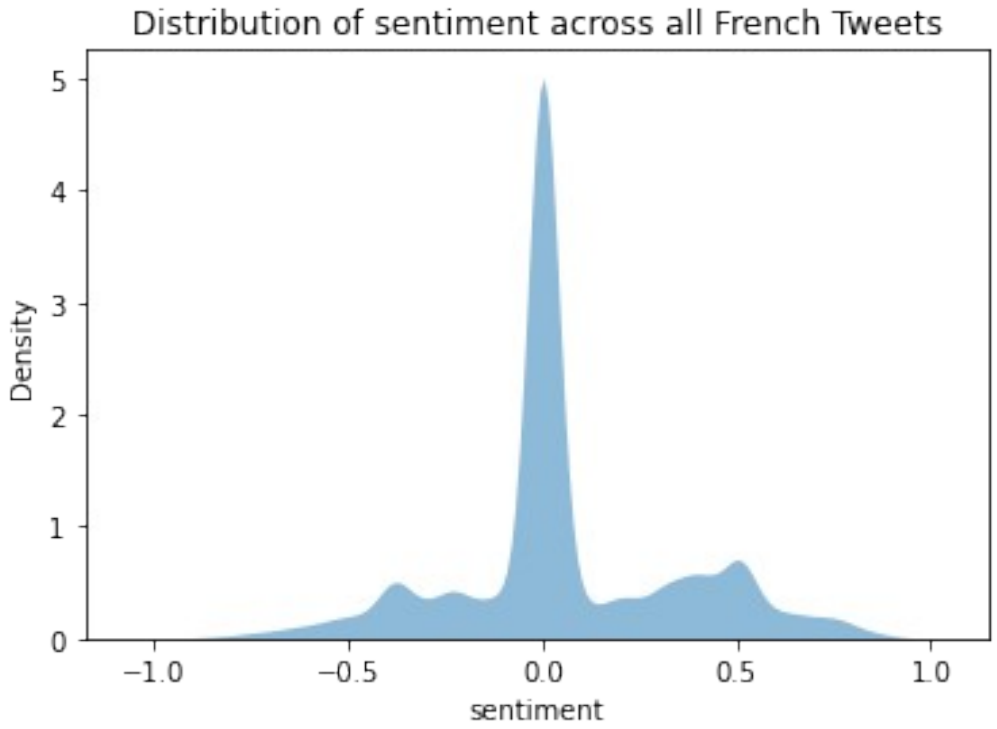
```
<matplotlib.image.AxesImage at 0x7f583351f430>
```

Like the previous two word clouds, "coronavirus" and "confin" stand out the most. Interesting enough, the french hashtag of "stay at home", "restezchezvous", was much more popular than the counterparts in English or Spanish.

We will plot KDE plots similar to before.

```
sns.kdeplot(data=french_numerics_df, x='sentiment',
            fill=True, alpha=0.5, linewidth = 0).set(
                title = "Distribution of sentiment across all French
Tweets"
            )
```

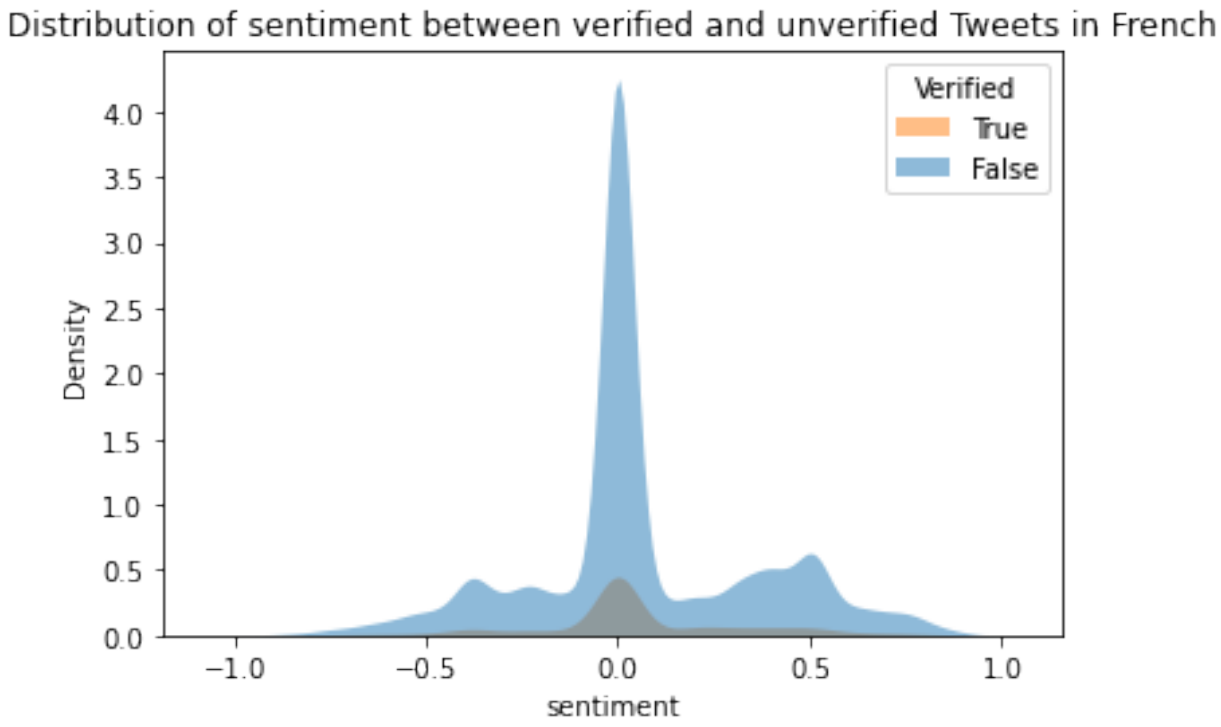
```
[Text(0.5, 1.0, 'Distribution of sentiment across all French Tweets')]
```



In this KDE plot we see majority of tweets have neutral sentiment. This phenomenon may be reflective of the tweets, but also maybe influenced by the quality of our sentiment analyzer.

```
sns.kdeplot(data = french_numerics_df, x='sentiment', hue='verified',
            fill=True, alpha=0.5, linewidth=0
            ).set(title='Distribution of sentiment between verified
and unverified Tweets in French')
plt.legend(title='Verified', labels=['True', 'False'])
```

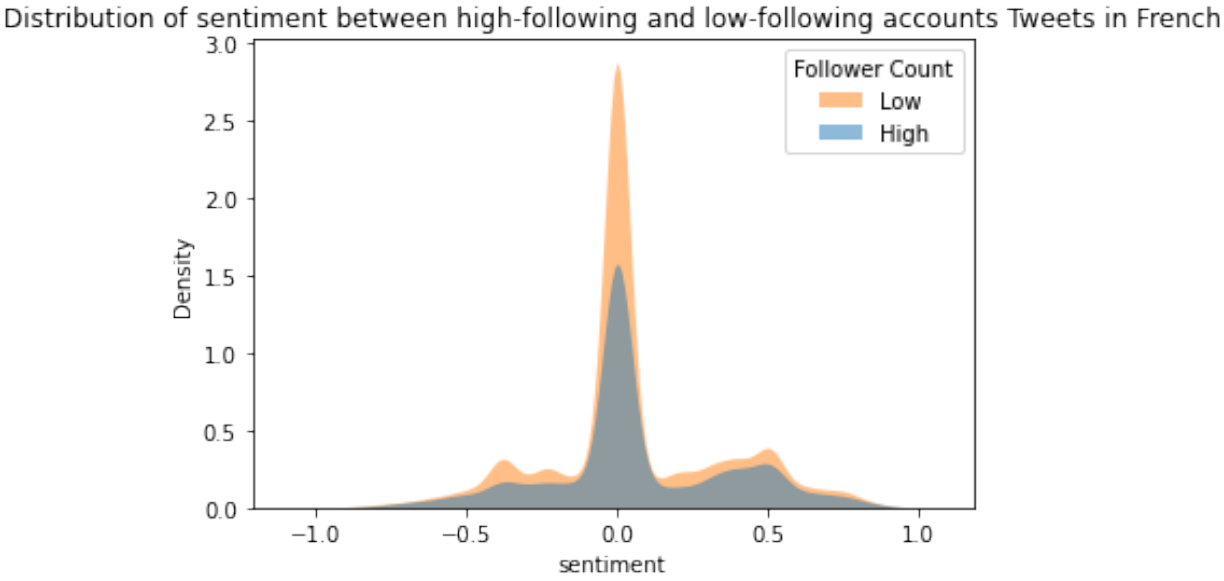
```
<matplotlib.legend.Legend at 0x7f580a3263d0>
```



In this distribution, verified and unverified tweets seem to take on the same distribution pattern, suggesting there are no significant trends.

```
sns.kdeplot(data = french_numerics_df, x='sentiment',
hue='follower_classification',
            fill=True, alpha=0.5, linewidth=0
            ).set(title='Distribution of sentiment between high-
following and low-following accounts Tweets in French')
plt.legend(title='Follower Count', labels=['Low', 'High'])
```

```
<matplotlib.legend.Legend at 0x7f580a296fd0>
```



In this distribution, high following and low following tweets seem to take on the same distribution pattern, suggesting there are no significant trends.

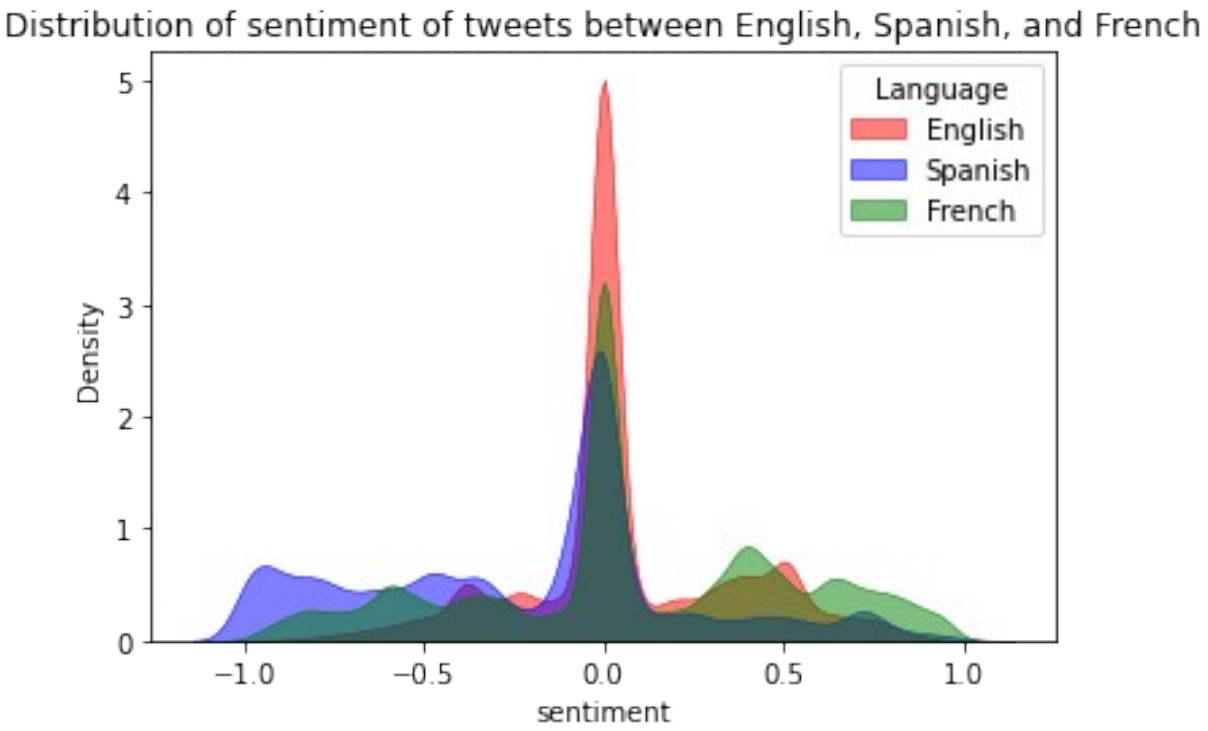
1.6 Visualizations for English, Spanish, and French

Now that we have the processed datasets, along with the sentiment scores for English, Spanish, and French, we want to plot graphs to show easy-to-understand comparisons between various factors. For example, we will be plotting KDE plots to compare sentiment between different languages, along with comparisons between verified vs non-verified sentiment tweets across the different languages.

First we plot sentiment for all languages to see how they compare.

```
g = sns.kdeplot(data = french_numerics_df, x='sentiment', color="red",
                fill=True, common_norm=False, palette="crest",
                alpha=0.5, linewidth=.5)
h = sns.kdeplot(data = spanish_numerics_df, x='sentiment',
color="blue",
            fill=True, common_norm=False, palette="crest",
                alpha=0.5, linewidth=.5)
i = sns.kdeplot(data = english_numerics_df, x='sentiment',
color="green",
            fill=True, common_norm=False, palette="crest",
                alpha=0.5, linewidth=.5).set(title="Distribution of
sentiment of tweets between English, Spanish, and French")
plt.legend(title='Language', loc='upper right', labels=['English',
'Spanish', 'French'])
```

```
<matplotlib.legend.Legend at 0x7f580a1446a0>
```



In this combined KDE plot of all three languages, we see that each language dominates in different regions of sentiment. English dominates neutral, Spanish in negative, and French in positive.

Next we plot aggregate distributions across languages for verification status and followers classification.

```
english_numerics_df['language'] = 'es'
french_numerics_df['language'] = 'fr'
spanish_numerics_df['language'] = 'es'
numerics_df = english_numerics_df
numerics_df.append(french_numerics_df)
numerics_df.append(spanish_numerics_df)
```

```
<ipython-input-169-276b7e516aa5>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  english_numerics_df['language'] = 'es'
<ipython-input-169-276b7e516aa5>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
```

```
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  french_numerics_df['language'] = 'fr'
<ipython-input-169-276b7e516aa5>:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  spanish_numerics_df['language'] = 'es'
```

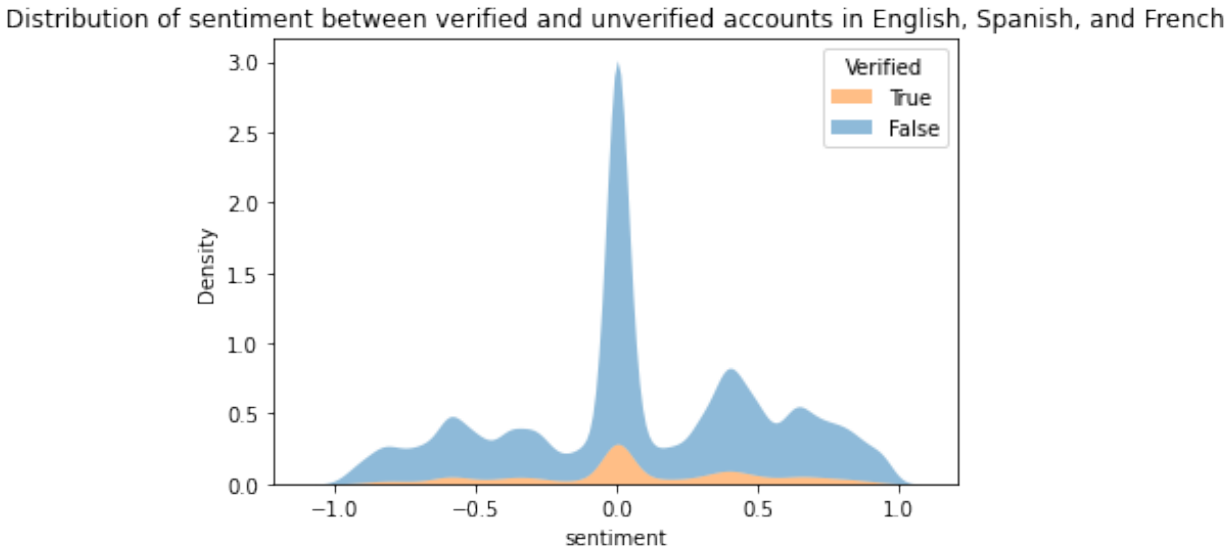
	favourites_count	retweet_count	follower_classification	\
8	16531	0	1	
10	2975	4	1	
13	495	0	1	
14	5450	6	1	
16	500	0	1	
...	...	...	...	
536129	75563	0	1	
536130	1322	0	1	
536131	6753	3	1	
536148	3886	1	1	
536155	26271	0	0	

	friends_count	verified	sentiment	language
8	2761	0	0.440400	es
10	125	1	0.000000	es
13	510	1	-0.296000	es
14	776	1	0.510600	es
16	2137	0	0.000000	es
...	...	...	...	...
536129	300	0	-0.700830	es
536130	380	0	0.329841	es
536131	1937	0	-0.797787	es
536148	126	0	-0.004215	es
536155	1214	0	0.169705	es

```
[342688 rows x 7 columns]
```

```
a = sns.kdeplot(data = numerics_df, x='sentiment', hue='verified',
                multiple="stack", alpha=0.5,
                linewidth=0).set(title="Distribution of sentiment
between verified and unverified accounts in English, Spanish, and
French")
plt.legend(title='Verified', loc='upper right', labels=['True',
'False'])
```

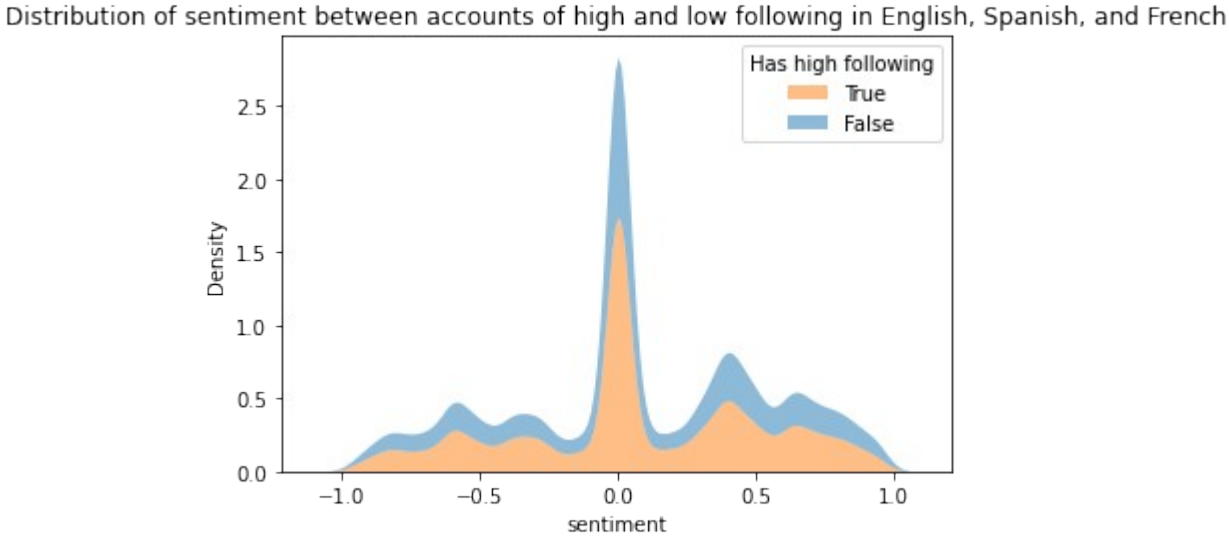
```
<matplotlib.legend.Legend at 0x7f580a25d400>
```



In this distribution, verified and unverified tweets seem to take on the same distribution pattern, suggesting there are no significant trends.

```
b = sns.kdeplot(data=numerics_df, x='sentiment',
hue='follower_classification',
                multiple='stack', alpha=0.5,
                linewidth=0).set(title = 'Distribution of sentiment
between accounts of high and low following in English, Spanish, and
French')
plt.legend(title='Has high following', loc='upper right',
labels=['True', 'False'])
```

```
<matplotlib.legend.Legend at 0x7f580a002f40>
```

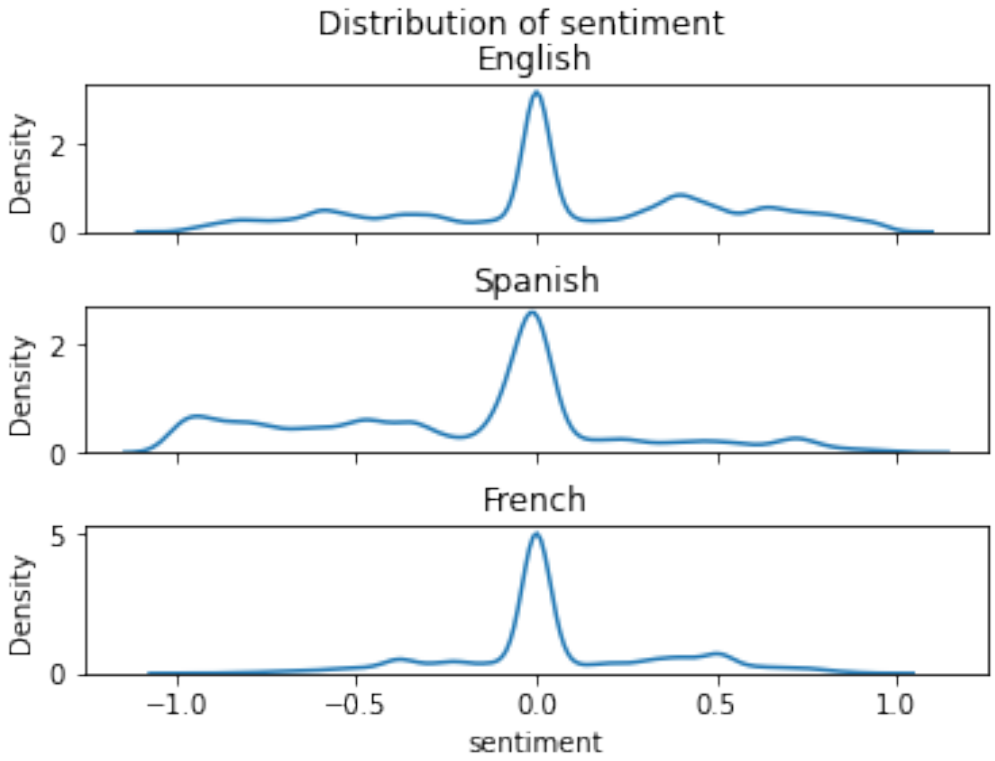


In this distribution, high and low following tweets seem to take on the same distribution pattern, suggesting there are no significant trends.

Next we will visualize the same numbers as above, except we will plot each language on its own individual subplot.

```
#Create subplots
fig, (ax1, ax2, ax3) = plt.subplots(nrows=3, sharex=True)
fig.subplots_adjust(hspace=0.5, wspace=0.125)

#Plot data by language
sns.kdeplot(data=english_numerics_df, x='sentiment', ax=ax1)
sns.kdeplot(data=spanish_numerics_df, x='sentiment', ax=ax2)
sns.kdeplot(data=french_numerics_df, x='sentiment', ax=ax3)

#Give each plot a title
ax1.set_title('English')
ax2.set_title('Spanish')
ax3.set_title('French')
fig.suptitle("Distribution of sentiment")
```

```
Text(0.5, 0.98, 'Distribution of sentiment')
```

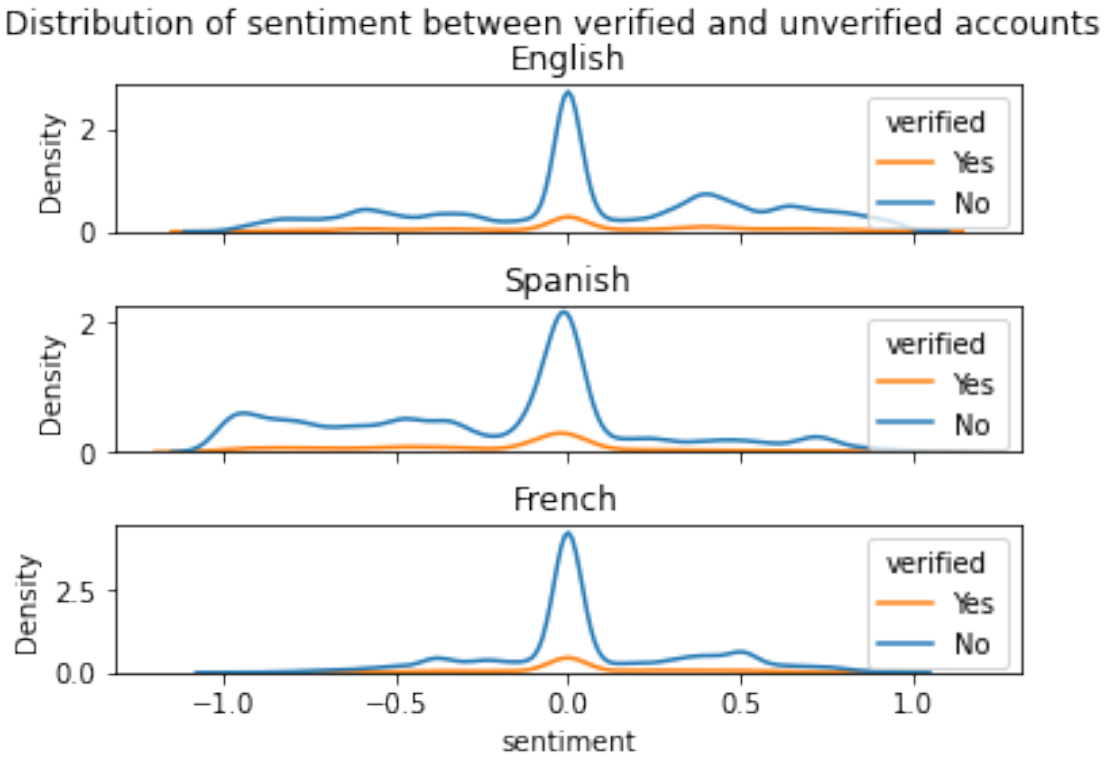


With the KDE plots side by side, we can see the differences in shape of each language's distributions. Similar to the conclusion drawn before, Spanish tweets have more negative sentiment than the other two languages.

```
#Create subplots
fig, (ax1, ax2, ax3) = plt.subplots(nrows=3, sharex=True)
```

```
fig.subplots_adjust(hspace=0.5, wspace=0.15)

#Plot data by language
sns.kdeplot(data=english_numerics_df, x='sentiment', hue='verified',
ax=ax1)
sns.kdeplot(data=spanish_numerics_df, x='sentiment', hue='verified',
ax=ax2)
sns.kdeplot(data=french_numerics_df, x='sentiment', hue='verified',
ax=ax3)

#Adjust the legend
ax1.legend(title="verified", loc='upper right', labels=['Yes', 'No'])
ax2.legend(title="verified", loc='upper right', labels=['Yes', 'No'])
ax3.legend(title="verified", loc='upper right', labels=['Yes', 'No'])

#Give each plot a title
ax1.set_title('English')
ax2.set_title('Spanish')
ax3.set_title('French')
fig.suptitle("Distribution of sentiment between verified and
unverified accounts")
```

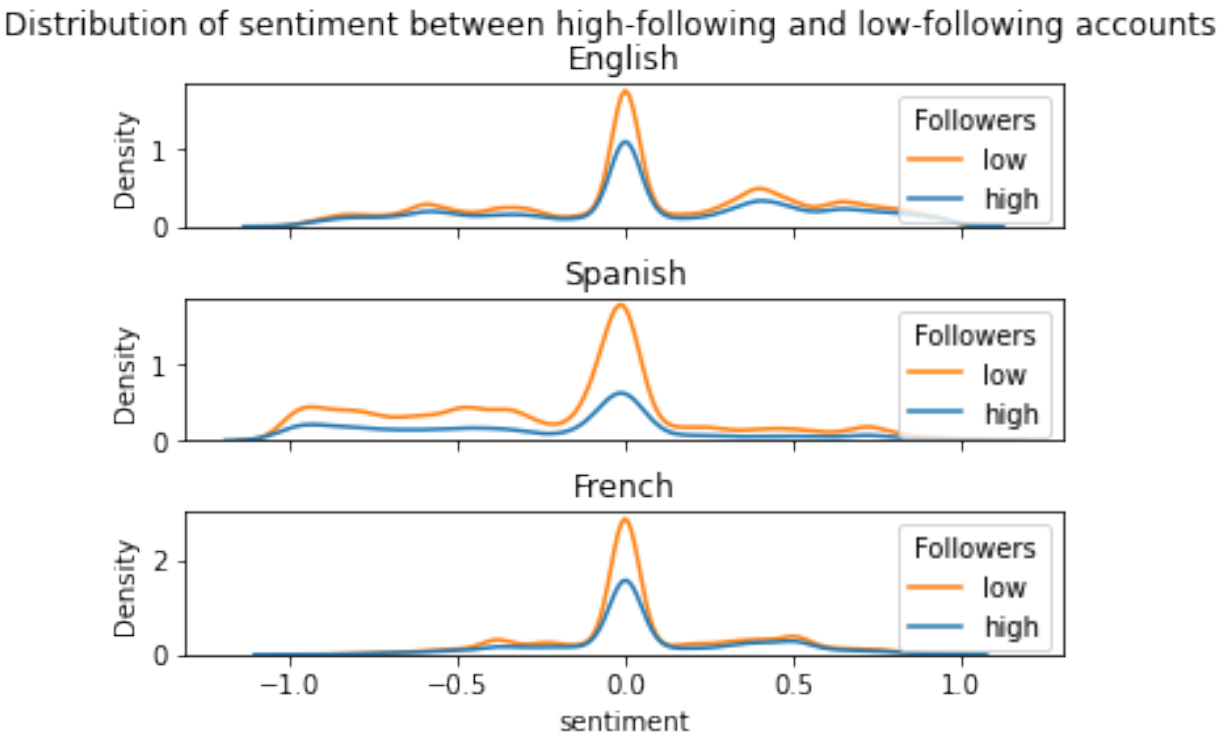
```
Text(0.5, 0.98, 'Distribution of sentiment between verified and
unverified accounts')
```



In this set of KDE plots, we can see that verified English tweets tend to be more positive while the other two languages stay fairly neutral.

```
#Create subplots
fig, (ax1, ax2, ax3) = plt.subplots(nrows=3, sharex=True)
fig.subplots_adjust(hspace=0.5, wspace=0.125)

#plot the data by language
sns.kdeplot(data=english_numerics_df, x='sentiment',
hue='follower_classification', ax=ax1)
sns.kdeplot(data=spanish_numerics_df, x='sentiment',
hue='follower_classification', ax=ax2)
sns.kdeplot(data=french_numerics_df, x='sentiment',
hue='follower_classification', ax=ax3)

#Adjust the legends
ax1.legend(title="Followers", loc='upper right', labels=['low',
'high'])
ax2.legend(title="Followers", loc='upper right', labels=['low',
'high'])
ax3.legend(title="Followers", loc='upper right', labels=['low',
'high'])

#Give each plot a title
ax1.set_title('English')
ax2.set_title('Spanish')
ax3.set_title('French')
fig.suptitle("Distribution of sentiment between high-following and
low-following accounts")
```

```
Text(0.5, 0.98, 'Distribution of sentiment between high-following and
low-following accounts')
```



In this distribution, the high and low following distributions per language stay very similar to the original distributions. We can conclude that regardless of language, follwer count is not a demographic that influences sentiment. This may be due to our tweets being derived from just one day, and thus more data could yield new results

Part II: Modeling

2.1 ZeroShot classification for labelling

We are going to be using a pretrained model trained on social media data to help generate labels for us to work with as the initial data was not labelled.

Dr. Wenpeng Yin proposed a method for using pre-trained NLI models as a ready-made zero-shot sequence classifiers. The method works by posing the sequence to be classified as the NLI premise and to construct a hypothesis from each candidate label. For example, if we want to evaluate whether a sequence belongs to the class "anxious", we could construct a hypothesis of This text is about anxiety.The probabilities for entailment and contradiction are then converted to label probabilities.

Reference: https://huggingface.co/facebook/bart-large-mnli

2.1.1 Initializing pipeline

We shall now import transformers and setup the nlp pipeline using the "facebook/bart-large-mnli" pretrained model for social media data

```
!pip install transformers
```

```
Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Collecting transformers
  Downloading transformers-4.25.1-py3-none-any.whl (5.8 MB)
ent already satisfied: regex!=2019.12.17 in
/usr/local/lib/python3.8/dist-packages (from transformers) (2022.6.2)
Requirement already satisfied: numpy>=1.17 in
/usr/local/lib/python3.8/dist-packages (from transformers) (1.21.6)
Requirement already satisfied: packaging>=20.0 in
/usr/local/lib/python3.8/dist-packages (from transformers) (21.3)
Collecting huggingface-hub<1.0,>=0.10.0
  Downloading huggingface_hub-0.11.1-py3-none-any.whl (182 kB)
ent already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.8/dist-
packages (from transformers) (6.0)
Requirement already satisfied: requests in
/usr/local/lib/python3.8/dist-packages (from transformers) (2.23.0)
Collecting tokenizers!=0.11.3,<0.14,>=0.11.1
  Downloading tokenizers-0.13.2-cp38-cp38-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (7.6 MB)
ent already satisfied: filelock in /usr/local/lib/python3.8/dist-
packages (from transformers) (3.8.0)
Requirement already satisfied: tqdm>=4.27 in
/usr/local/lib/python3.8/dist-packages (from transformers) (4.64.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/usr/local/lib/python3.8/dist-packages (from huggingface-
hub<1.0,>=0.10.0->transformers) (4.4.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/usr/local/lib/python3.8/dist-packages (from packaging>=20.0-
>transformers) (3.0.9)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1
in /usr/local/lib/python3.8/dist-packages (from requests-
>transformers) (1.24.3)
Requirement already satisfied: certifi>=2017.4.17 in
/usr/local/lib/python3.8/dist-packages (from requests->transformers)
(2022.9.24)
Requirement already satisfied: idna<3,>=2.5 in
/usr/local/lib/python3.8/dist-packages (from requests->transformers)
(2.10)
Requirement already satisfied: chardet<4,>=3.0.2 in
/usr/local/lib/python3.8/dist-packages (from requests->transformers)
(3.0.4)
Installing collected packages: tokenizers, huggingface-hub,
transformers
Successfully installed huggingface-hub-0.11.1 tokenizers-0.13.2
transformers-4.25.1
```

```
import torch
torch.cuda.empty_cache()
```

```
device = "cuda:0" if torch.cuda.is_available() else "cpu"

# pose sequence as a NLI premise and label as a hypothesis
from transformers import AutoModelForSequenceClassification,
AutoTokenizer, pipeline
nli_model =
AutoModelForSequenceClassification.from_pretrained("facebook/bart-
large-mnli")
tokenizer = AutoTokenizer.from_pretrained("facebook/bart-large-mnli")

nlp = pipeline("zero-shot-classification", model=nli_model,
tokenizer=tokenizer,  device=0)
```


```
{"model_id":"2ed4eccc01ee41dd92f9b31dfa8cd77e","version_major":2,"vers
ion_minor":0}

{"model_id":"b6e510f4f3de492ab5d84f99dd87050b","version_major":2,"vers
ion_minor":0}

{"model_id":"f68a2f42142b42e299fa5a500d681152","version_major":2,"vers
ion_minor":0}

{"model_id":"976724a42213428f8a5c569d4783c474","version_major":2,"vers
ion_minor":0}

{"model_id":"fe6ab84db4ae4a1eaed5724ef96ad53b","version_major":2,"vers
ion_minor":0}

{"model_id":"c55fa423c09c414eb3400c66e97b08a3","version_major":2,"vers
ion_minor":0}
```


Initialize the candidate labels we shall check against. In this case we want to look for depression inducing behaviours and hence have used the following labels.

With a better GPU and the ability to run over more data, we can incorporate more behavorial patterns such as "calmness" and many more.

Currently we are running over 6000 English tweets.

```
candidate_labels = ["sad", "relief", "anxious"]

english_tweets_df_reduced = english_tweets_df.head(6000)
english_tweets_df_reduced =
english_tweets_df_reduced.reset_index(drop=True)
```

2.1.2 Running ZeroShot Learning on the data to get Labels

We shall be storing the results into a final dataframe which shall be used for modelling in the next steps

```
from tqdm.auto import tqdm
hypothesis_template = "{}"
BATCH_SIZE = 32
scores = []
labels = []
hypothesisOutput = {}

for i in tqdm(range(0,
len(english_tweets_df_reduced['training_text'].to_list()),
BATCH_SIZE)):
  examples = english_tweets_df_reduced['training_text'].to_list()
[i:i+BATCH_SIZE]
  outputs = nlp(examples, candidate_labels, multi_label=True,
hypothesis_template="{}")
  scores.extend([o['scores'][0] for o in outputs])
  labels.extend([o['labels'][0] for o in outputs])
hypothesisOutput[f'Label'] = labels
hypothesisOutput[f'Score'] = scores
```

{"model_id":"2008fc5ebc7f49ae8c9a24b6e6bfc3a0","version_major":2,"version_minor":0}

```
/usr/local/lib/python3.8/dist-packages/transformers/pipelines/
base.py:1043: UserWarning: You seem to be using the pipelines
sequentially on GPU. In order to maximize efficiency please use a
dataset
  warnings.warn(
```

```
df_hypothesis = pd.DataFrame(hypothesisOutput)
temp_df = english_tweets_df_reduced[['training_text']]
final_df = pd.concat([temp_df, df_hypothesis], axis=1)

test_df = final_df
```

2.2 Multi-Label Text Classification

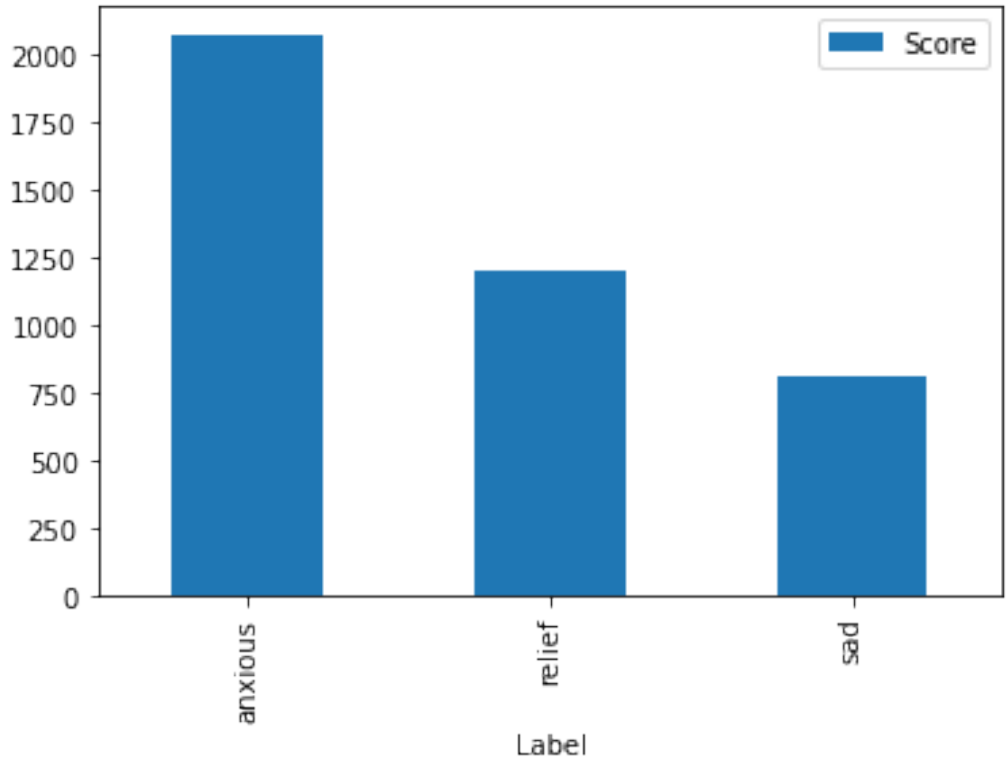
We have a reasonable distribution and shall use the data in a one hot encoded format to do Multi-Label Text Classification.

We shall be using the following ML methods:

1. Decision Tree Classifier
2. Random Forest with GridSearchCV for hyperparameter tuning
3. One vs Rest Classifier with accuracies for each category
4. Binary Relevance Classifier (Sub-optimal for multi-class)
5. Classifier Chains
6. Label Powerset Classifier
7. LSTM for multi-label classification

```
df_groups = test_df.groupby(['Label']).sum()

#create bar plot by group
df_groups.plot(kind='bar')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5721686b80>
```




```
test_df = pd.get_dummies(test_df, columns=['Label'], prefix='',
prefix_sep='', drop_first=False)
test_df['Label'] = final_df['Label']
```

2.2.1 Various ML Techniques for Multi-Label Text Classification

2.2.1.1 Initialize the data for the ML models

We are going to be using the TfidfVectorizer to help us convert the text we have into a vectorized format so that the machine will be able to understand the text in a tensor/array format of numbers based on a corpus of words created during the ***vectorizer.fit***

The ***vectorizer.transform*** coverts the training text which we preprocessed to be a clean string of feature words.

We are using the ***train_test_split*** method to separate the data into a 80% train and 20% test dataset. We are using the ***stratify_labels*** method to help make sure that the train and test dataset are split uniformly based on the candidate labels.

```
from sklearn.model_selection import train_test_split

tag_labels = test_df[candidate_labels]
train, test = train_test_split(test_df, random_state=42,
test_size=0.20, shuffle=True, stratify=tag_labels)

from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer(strip_accents='unicode', analyzer='word',
ngram_range=(1,3), norm='l2')
vectorizer.fit(train['training_text'])
vectorizer.fit(test['training_text'])

x_train = vectorizer.transform(train['training_text'])
y_train = train.drop(labels = ['Score', 'Label', 'training_text'],
axis=1)

x_test = vectorizer.transform(test['training_text'])
y_test = test.drop(labels = ['Score', 'Label', 'training_text'],
axis=1)
```

2.2.1.2 Decision Tree Classifier

Decision Tree is a Supervised Machine Learning Algorithm that uses a set of rules to make decisions, similarly to how humans make decisions.

One way to think of a Machine Learning classification algorithm is that it is built to make decisions.

You usually say the model predicts the class of the new, never-seen-before input but, behind the scenes, the algorithm has to decide which class to assign.

We see a reasonable accuracy and results based on the limitations in th dataset and the labels we are using

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
import seaborn as sns

dtc = DecisionTreeClassifier()
dtc.fit(x_train, y_train)
predictions = dtc.predict(x_test)



print("Accuracy = ",accuracy_score(y_test, predictions))


predictions = pd.DataFrame(predictions)
predictions = predictions.rename(columns={0:'anxious', 1:'relief',
```

```
2:'sad'})

from sklearn.metrics import classification_report
print('\nClassification Report\n')
print(classification_report(y_test.idxmax(axis=1),
predictions.idxmax(axis=1), target_names=candidate_labels))
```

```
Accuracy =  0.5241666666666667

Classification Report

              precision    recall  f1-score   support

         sad       0.53      0.84      0.65       594
      relief       0.55      0.28      0.37       374
     anxious       0.33      0.11      0.16       232

    accuracy                           0.52      1200
   macro avg       0.47      0.41      0.40      1200
weighted avg       0.50      0.52      0.47      1200
```

2.2.1.3 Random Forest Classifier

Usually, we only have a vague idea of the best hyperparameters and thus the best approach to narrow our search is to evaluate a wide range of values for each hyperparameter. Using Scikit-Learn's ***GridSearchCV*** method, we can define a grid of hyperparameter ranges, and sample from the grid, performing K-Fold CV with each combination of values.

In the end we get a list of optimal hyperparameters for the Random Forest Classifier.

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import GridSearchCV

rfc = RandomForestClassifier()
grid_params = {
    'n_estimators' : [90,120],
    'criterion' : ['gini','entropy'],
    'max_depth' : [2, 5, 10],
    'min_samples_leaf' : [1, 5, 10],
    'min_samples_split' : [2, 5, 10],
    'max_features' : ['auto', 'log2']
}

grid_search = GridSearchCV(estimator=rfc, param_grid=grid_params,
cv=5, n_jobs=-1, verbose=3)

grid_search.fit(x_train, y_train)

grid_search.best_params_
```

```
Fitting 5 folds for each of 216 candidates, totalling 1080 fits

{'criterion': 'gini',
 'max_depth': 10,
 'max_features': 'auto',
 'min_samples_leaf': 1,
 'min_samples_split': 10,
 'n_estimators': 120}
```

Use the optimal hyperparameters as mentioned above for our Random Forest Classifier

```
rfc = RandomForestClassifier(criterion=
grid_search.best_params_['criterion'],
 max_depth = grid_search.best_params_['max_depth'],
 max_features= grid_search.best_params_['max_features'],
 min_samples_leaf = grid_search.best_params_['min_samples_leaf'],
 min_samples_split = grid_search.best_params_['min_samples_split'],
 n_estimators = grid_search.best_params_['n_estimators'])
rfc.fit(x_train, y_train)
predictions = rfc.predict(x_test)

print("Accuracy = ",accuracy_score(y_test, predictions))

Accuracy =  0.4716666666666667
```

2.2.1.4 One vs Rest Classifier

The Multi-label algorithm accepts a binary mask over multiple labels. The result for each prediction will be an array of 0s and 1s marking which class labels apply to each row input sample.

Logistic Regression

OneVsRest strategy can be used for multi-label learning, where a classifier is used to predict multiple labels for instance. Logistic Regression supports multi-class, but we are in a multi-label scenario, therefore, we wrap *Logistic Regression in the OneVsRestClassifier.

```
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
from sklearn.metrics import accuracy_score
from sklearn.multiclass import OneVsRestClassifier
# Using pipeline for applying logistic regression and one vs rest
classifier
LogReg_pipeline = Pipeline([
                ('clf',
OneVsRestClassifier(LogisticRegression(solver='sag'), n_jobs=-1)),
            ])

categories = candidate_labels
```

```
for category in categories:
    print('**Processing {} tweets...**'.format(category))

    # Training logistic regression model on train data
    LogReg_pipeline.fit(x_train, y_train[category])

    # calculating test accuracy
    prediction = LogReg_pipeline.predict(x_test)
    print('Test accuracy is
{}'.format(accuracy_score(y_test[category], prediction)))
    print("\n")
```

```
**Processing sad tweets...**
Test accuracy is 0.8066666666666666


**Processing relief tweets...**
Test accuracy is 0.7216666666666667


**Processing anxious tweets...**
Test accuracy is 0.6925
```

2.2.1.5 Binary Relevance Classifier

In this case an ensemble of single-label binary classifiers is trained, one for each class. Each classifier predicts either the membership or the non-membership of one class. The union of all classes that were predicted is taken as the multi-label output. This approach is popular because it is easy to implement, however it also ignores the possible correlations between class labels.

This is suboptimal for our scenario as it ignored correlations and is better for Binary Cases.

First we install ***scikit-multilearn*** and then implement

```
!pip install scikit-multilearn
```

```
Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Collecting scikit-multilearn
  Downloading scikit_multilearn-0.2.0-py3-none-any.whl (89 kB)
ultilearn
Successfully installed scikit-multilearn-0.2.0
```

```
# using binary relevance
from skmultilearn.problem_transform import BinaryRelevance
from sklearn.naive_bayes import GaussianNB
# initialize binary relevance multi-label classifier
# with a gaussian naive bayes base classifier
classifier = BinaryRelevance(GaussianNB())
```

```
# train
classifier.fit(x_train, y_train)
# predict
predictions = classifier.predict(x_test)
# accuracy
print("Accuracy = ",accuracy_score(y_test,predictions))
```

```
Accuracy =  0.4325
```

2.2.1.6 Classifier Chain

A chain of binary classifiers C0, C1, . . . , Cn is constructed, where a classifier Ci uses the predictions of all the classifier Cj , where j < i. This way the method, also called classifier chains (CC), can take into account label correlations.

The total number of classifiers needed for this approach is equal to the number of classes, but the training of the classifiers is more involved.

Following is an illustrated example with a classification problem of three categories {C1, C2, C3} chained in that order.

This works really well for our scenario.

```
# using classifier chains
from skmultilearn.problem_transform import ClassifierChain
from sklearn.linear_model import LogisticRegression
# initialize classifier chains multi-label classifier
classifier = ClassifierChain(LogisticRegression())
# Training logistic regression model on train data
classifier.fit(x_train, y_train)
# predict
predictions = classifier.predict(x_test)
# accuracy
print("Accuracy = ",accuracy_score(y_test,predictions))
print("\n")
```

```
Accuracy =  0.6166666666666667
```

2.2.1.7 Label Powerset Classifier

This approach does take possible correlations between class labels into account. More commonly this approach is called the label-powerset method, because it considers each member of the power set of labels in the training set as a single label.

This method needs worst case (2^|C|) classifiers, and has a high computational complexity.

However when the number of classes increases the number of distinct label combinations can grow exponentially. This easily leads to combinatorial explosion and thus computational infeasibility. Furthermore, some label combinations will have very few positive examples.

This would be sub-optimal in the case that we have an increased number of behavior classes and can cause computational explosion

```
# using Label Powerset
from skmultilearn.problem_transform import LabelPowerset
# initialize label powerset multi-label classifier
classifier = LabelPowerset(LogisticRegression())
# train
classifier.fit(x_train, y_train)
# predict
predictions = classifier.predict(x_test)
# accuracy
print("Accuracy = ",accuracy_score(y_test,predictions))
print("\n")
```

```
/usr/local/lib/python3.8/dist-packages/sklearn/linear_model/
_logistic.py:814: ConvergenceWarning: lbfgs failed to converge
(status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as
shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  n_iter_i = _check_optimize_result(

Accuracy =  0.6066666666666667
```

2.2.2 Implementing an LSTM to potentially increase performance

2.2.2.1 Initialising the LSTM

Vectorize tweets text, by turning each text into either a sequence of integers or into a vector.

Limit the data set to the top 15,000 words.

Set the max number of words in each tweet at 50. This is most of what we see when it comes to number of words used per tweet

First we shall look at the distribution of the labels.

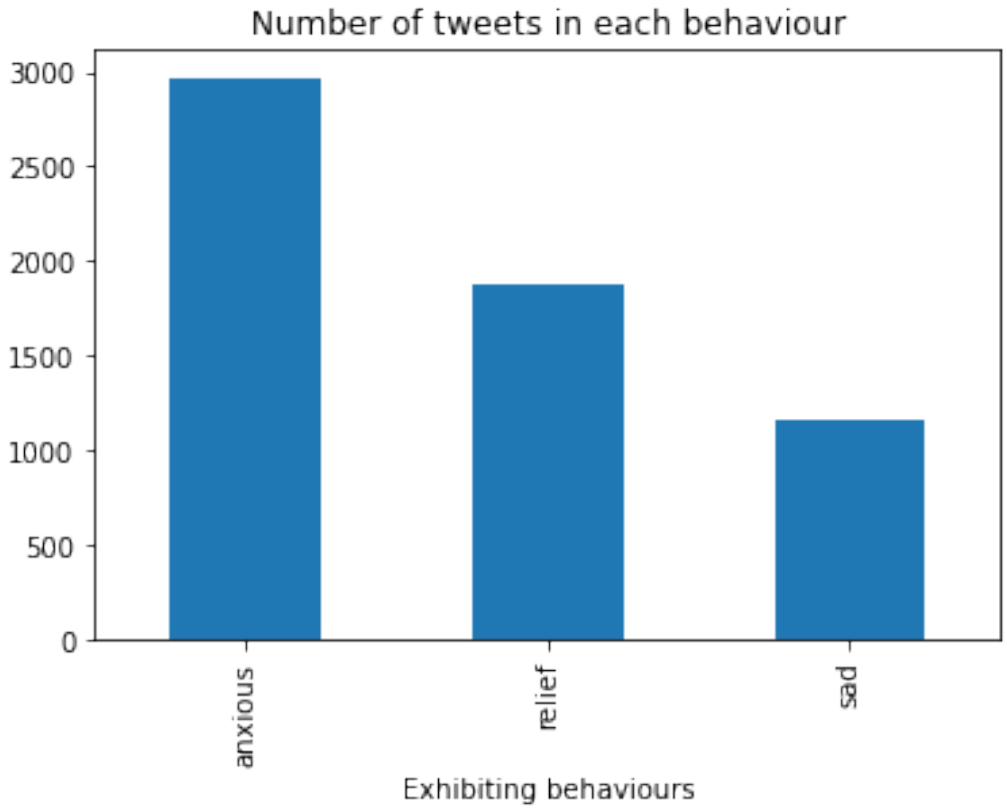
We are using the LSTM to potentially improve the performance of the Classifier on unseen tweets for behavior extraction labels.

```
test_df.Label.value_counts()
```

```
anxious    2967
relief     1872
sad        1161
Name: Label, dtype: int64
```

```
test_df['Label'].value_counts().sort_values(ascending=False).plot(kind
='bar', xlabel='Exhibiting behaviours',

title='Number of tweets in each behaviour')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5721633580>
```



Recurrent Neural Network (RNN) using the Long Short Term Memory (LSTM) architecture can be implemented using Keras.

Reference to Keras library: https://keras.io/

```
from keras_preprocessing.text import Tokenizer
from keras_preprocessing.sequence import pad_sequences
from keras.models import Sequential
from keras.layers import Dense, Embedding, LSTM, SpatialDropout1D
from sklearn.model_selection import train_test_split
from keras.utils.np_utils import to_categorical
from keras.callbacks import EarlyStopping
from keras.layers import Dropout
```

```
# The maximum number of words to be used. (most frequent)
MAX_NB_WORDS = 15000
# Max number of words in each tweet.
MAX_SEQUENCE_LENGTH = 50
# This is fixed.
EMBEDDING_DIM = 100

tokenizer = Tokenizer(num_words=MAX_NB_WORDS, filters='!"#$
%&()*+,-./:;<=>?@[\]^_`{|}~', lower=True)
tokenizer.fit_on_texts(test_df['training_text'].values)
word_index = tokenizer.word_index
print('Found %s unique tokens.' % len(word_index))
```

```
Found 25561 unique tokens.
```

Truncate and pad the input sequences so that they are all in the same length for modeling.

```
X = tokenizer.texts_to_sequences(test_df['training_text'].values)
X = pad_sequences(X, maxlen=MAX_SEQUENCE_LENGTH)
print('Shape of data tensor:', X.shape)
```

```
Shape of data tensor: (6000, 50)
```

Use the one hot encoded categorical labels as the target

```
Y= test_df[candidate_labels]
print('Shape of label tensor:', Y.shape)
```

```
Shape of label tensor: (6000, 3)
```

Train and Test Split (80/20 respectively)

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size =
0.20, random_state = 42)
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(4800, 50) (4800, 3)
(1200, 50) (1200, 3)
```

The first layer is the embedded layer that uses 100 length vectors to represent each word.

SpatialDropout1D performs variational dropout in NLP models.

The next layer is the LSTM layer with 100 memory units.

The output layer must create 3 output values, one for each class.

Activation function is softmax for multi-class classification.

Because it is a multi-class classification problem, categorical_crossentropy is used as the loss function.

```
model = Sequential()
model.add(Embedding(MAX_NB_WORDS, EMBEDDING_DIM,
input_length=X.shape[1]))
model.add(SpatialDropout1D(0.2))
model.add(LSTM(100, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(3, activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])
print(model.summary())
```

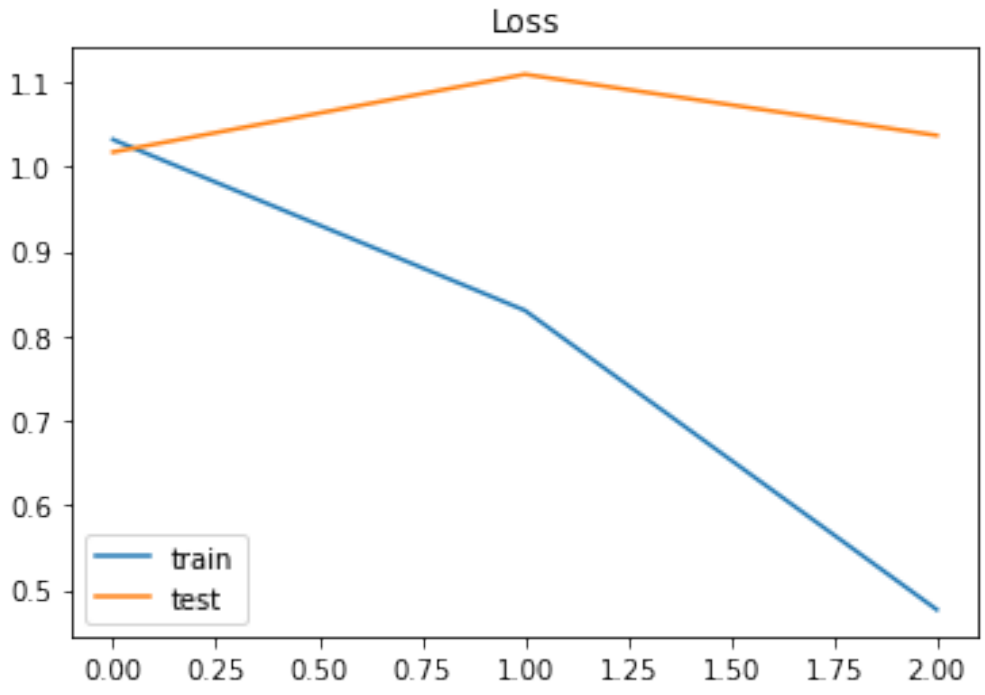
```
WARNING:tensorflow:Layer lstm will not use cuDNN kernels since it
doesn't meet the criteria. It will use a generic GPU kernel as
fallback when running on GPU.

Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 50, 100)           1500000

 spatial_dropout1d (SpatialD  (None, 50, 100)          0
 ropout1D)

 lstm (LSTM)                 (None, 100)               80400

 dense (Dense)               (None, 3)                 303

=================================================================
Total params: 1,580,703
Trainable params: 1,580,703
Non-trainable params: 0
_________________________________________________________________
None
```

We reduced the number of epochs to help improve accuracy and to reduce overfitting. We shall see the plot referenced in the futre

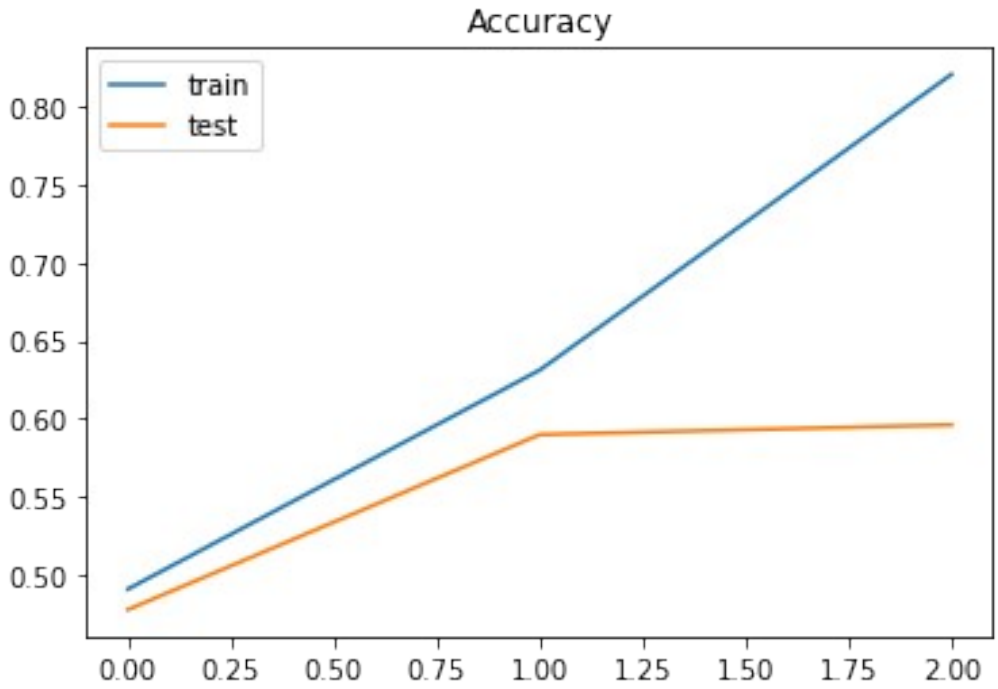
```
epochs = 3
batch_size = 64

history = model.fit(X_train, Y_train, epochs=epochs,
batch_size=batch_size,validation_split=0.1,callbacks=[EarlyStopping(mo
nitor='val_loss', patience=3, min_delta=0.0001)])
```

```
Epoch 1/3
68/68 [==============================] - 18s 229ms/step - loss: 1.0322
- accuracy: 0.4903 - val_loss: 1.0174 - val_accuracy: 0.4771
```

```
Epoch 2/3
68/68 [==============================] - 16s 221ms/step - loss: 0.8302
- accuracy: 0.6310 - val_loss: 1.1093 - val_accuracy: 0.5896
Epoch 3/3
68/68 [==============================] - 19s 285ms/step - loss: 0.4764
- accuracy: 0.8211 - val_loss: 1.0371 - val_accuracy: 0.5958

accr = model.evaluate(X_test,Y_test)
print('Test set\n  Loss: {:0.3f}\n  Accuracy:
{:0.3f}'.format(accr[0],accr[1]))

38/38 [==============================] - 1s 17ms/step - loss: 1.0114 -
accuracy: 0.6058
Test set
  Loss: 1.011
  Accuracy: 0.606

plt.title('Loss')
plt.plot(history.history['loss'], label='train')
plt.plot(history.history['val_loss'], label='test')
plt.legend()
plt.show();
```




```
plt.title('Accuracy')
plt.plot(history.history['accuracy'], label='train')
plt.plot(history.history['val_accuracy'], label='test')
plt.legend()
plt.show();
```



The plots suggest that the model has a little over fitting problem, more data may help, but more epochs will not help using the current data.

2.2.2.2 Inferences based on example tweets

The tweets used show that the model can give good performance in pulling out behaviours. The test and training data may not be optimal but considering the groun truths used and the overall complexity of NLP modelling the model has great performance on Unseen tweets

Example for tweet showing anxious behavior

```
new_tweet = ['I am so worried about the current state of the covid
vaccine repurcussions, I feel unsure whether to take the vaccine or
not']
seq = tokenizer.texts_to_sequences(new_tweet)
padded = pad_sequences(seq, maxlen=MAX_SEQUENCE_LENGTH)
pred = model.predict(padded)
labels = candidate_labels
print(pred, 'Label predicted by model is:',labels[np.argmax(pred)])
```

```
1/1 [==============================] - 0s 312ms/step
[[0.00447803 0.00547479 0.99004716]] Label predicted by model is:
anxious
```

Example for tweet showing relieved behavior

```
new_tweet = ['Vaccine companies say a continued focus on getting the
vaccine ready is making me feel relief']
seq = tokenizer.texts_to_sequences(new_tweet)
```

```
padded = pad_sequences(seq, maxlen=MAX_SEQUENCE_LENGTH)
pred = model.predict(padded)
labels = candidate_labels
print(pred, 'Label predicted by model is:',labels[np.argmax(pred)])
```

```
1/1 [==============================] - 0s 26ms/step
[[0.02965348 0.84835696 0.12198953]] Label predicted by model is:
relief
```

Example for tweet showing sad behavior

```
new_tweet = ['I am feeling sad right now with the current state of
covid']
seq = tokenizer.texts_to_sequences(new_tweet)
padded = pad_sequences(seq, maxlen=MAX_SEQUENCE_LENGTH)
pred = model.predict(padded)
labels = candidate_labels
print(pred, 'Label predicted by model is:',labels[np.argmax(pred)])
```

```
1/1 [==============================] - 0s 33ms/step
[[0.6191316  0.1001002  0.28076816]] Label predicted by model is: sad
```

2.3 ZeroShot classification for French labelling

We are going to be using a pretrained multi-lingual model trained on social media data to help generate labels for us to work with as the initial Italian data was not labelled.

This model is intended to be used for zero-shot text classification, especially in languages other than English. It is fine-tuned on XNLI, which is a multilingual NLI dataset. The model can therefore be used with any of the languages in the XNLI corpus:

English French Spanish German Greek Bulgarian Russian Turkish Arabic Vietnamese Thai Chinese Hindi Swahili Urdu

Since the base model was pre-trained trained on 100 different languages, the model has shown some effectiveness in languages beyond those listed above as well. See the full list of pre-trained languages in appendix A of the XLM Roberata paper

Reference: https://huggingface.co/joeddav/xlm-roberta-large-xnli XLM Roberata paper: https://arxiv.org/abs/1911.02116

2.3.1 Initializing pipeline

We shall now import transformers and setup the nlp pipeline using the "joeddav/xlm-roberta-large-xnli" multi-lingual pretrained model for social media data

```
!pip install sentencepiece
```

```
Looking in indexes: https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/
Collecting sentencepiece
```

```
  Downloading sentencepiece-0.1.97-cp38-cp38-
manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.3 MB)
```

```
import torch
torch.cuda.empty_cache()
device = "cuda:0" if torch.cuda.is_available() else "cpu"

# pose sequence as a NLI premise and label as a hypothesis
from transformers import AutoModelForSequenceClassification,
AutoTokenizer, pipeline
# nli_model =
AutoModelForSequenceClassification.from_pretrained("joeddav/xlm-
roberta-large-xnli")
# tokenizer = AutoTokenizer.from_pretrained("joeddav/xlm-roberta-
large-xnli")
nlp = pipeline("zero-shot-classification", model="joeddav/xlm-roberta-
large-xnli", device=0)

# nlp = pipeline("zero-shot-classification", model=nli_model,
tokenizer=tokenizer,  device=0)
```


```
{"model_id":"23433126a02a4c3cb5d381029f6db06f","version_major":2,"vers
ion_minor":0}

{"model_id":"bd69a36741054b7cbe8864cf5ac360c1","version_major":2,"vers
ion_minor":0}
```


```
Some weights of the model checkpoint at joeddav/xlm-roberta-large-xnli
were not used when initializing XLMRobertaForSequenceClassification:
['roberta.pooler.dense.weight', 'roberta.pooler.dense.bias']
- This IS expected if you are initializing
XLMRobertaForSequenceClassification from the checkpoint of a model
trained on another task or with another architecture (e.g.
initializing a BertForSequenceClassification model from a
BertForPreTraining model).
- This IS NOT expected if you are initializing
XLMRobertaForSequenceClassification from the checkpoint of a model
that you expect to be exactly identical (initializing a
BertForSequenceClassification model from a
BertForSequenceClassification model).
```


```
{"model_id":"d295cd94e3a746569f3401a20dea1326","version_major":2,"vers
ion_minor":0}

{"model_id":"09cc9c0b07d84e659333d7d49eb170aa","version_major":2,"vers
ion_minor":0}

{"model_id":"df44c5a0572f45eabdec3f317d3ad766","version_major":2,"vers
ion_minor":0}
```

Initialize the candidate labels we shall check against. In this case we want to look for depression inducing behaviours and hence have used the following labels.

With a better GPU and the ability to run over more data, we can incorporate more behavorial patterns such as "calmness" and many more.

Currently we are running over 6000 French tweets.

```
candidate_labels = ["sad", "relief", "anxious"]

french_tweets_df_reduced = french_tweets_df.head(6000)
french_tweets_df_reduced =
french_tweets_df_reduced.reset_index(drop=True)
```

2.3.2 Running ZeroShot Learning on the data to get Labels

We shall be storing the results into a final dataframe which shall be used for modelling in the next steps

```
from tqdm.auto import tqdm
hypothesis_template = "{}"
BATCH_SIZE = 32
scores = []
labels = []
hypothesisOutput = {}

for i in tqdm(range(0,
len(english_tweets_df_reduced['training_text'].to_list()),
BATCH_SIZE)):
  examples = english_tweets_df_reduced['training_text'].to_list()
[i:i+BATCH_SIZE]
  outputs = nlp(examples, candidate_labels, multi_label=True,
hypothesis_template="{}")
  scores.extend([o['scores'][0] for o in outputs])
  labels.extend([o['labels'][0] for o in outputs])
hypothesisOutput[f'Label'] = labels
hypothesisOutput[f'Score'] = scores
```

```
{"model_id":"7008371543284905bf656527e65e607c","version_major":2,"vers
ion_minor":0}
```

```
/usr/local/lib/python3.8/dist-packages/transformers/pipelines/
base.py:1043: UserWarning: You seem to be using the pipelines
sequentially on GPU. In order to maximize efficiency please use a
dataset
  warnings.warn(
```

```
df_hypothesis = pd.DataFrame(hypothesisOutput)
temp_df = french_tweets_df_reduced[['training_text']]
final_french_df = pd.concat([temp_df, df_hypothesis], axis=1)
```

```
test_french_df = final_french_df
```

2.4 Multi-Label Text French Classification

We have a reasonable distribution and shall use the data in a one hot encoded format to do Multi-Label Text Classification.

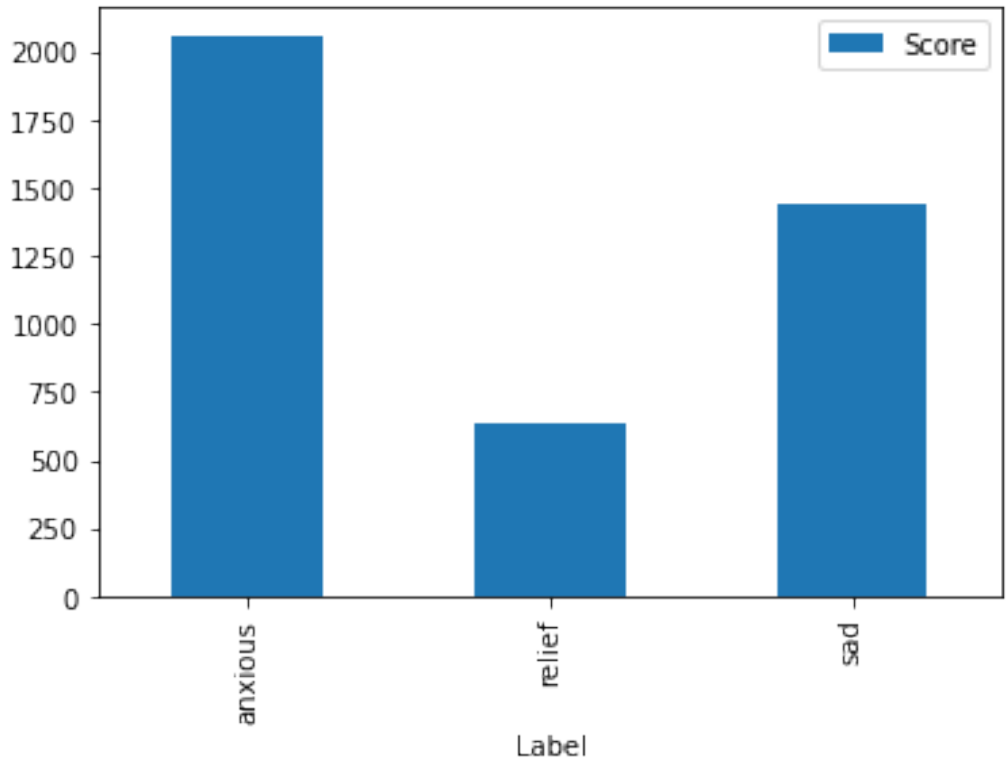
We shall be using the following ML methods:

1. Decision Tree Classifier
2. Random Forest with GridSearchCV for hyperparameter tuning
3. LSTM for multi-label classification

We can use all the same methods mentioned above including the LSTM for the French data. We would like to showcase a few of the methods as an example

```
df_groups = test_french_df.groupby(['Label']).sum()

#create bar plot by group
df_groups.plot(kind='bar')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f57188deb20>
```



```
test_french_df = pd.get_dummies(test_df, columns=['Label'], prefix='',
prefix_sep='', drop_first=False)
test_french_df['Label'] = final_df['Label']
```

2.2.1 Various ML Techniques for Multi-Label Text Classification

2.4.1.1 Initialize the data for the ML models for French Data

We are going to be using the TfidfVectorizer to help us convert the text we have into a vectorized format so that the machine will be able to understand the text in a tensor/array format of numbers based on a corpus of words created during the ***vectorizer.fit***

The ***vectorizer.transform*** coverts the training text which we preprocessed to be a clean string of feature words.

We are using the ***train_test_split*** method to separate the data into a 80% train and 20% test dataset. We are using the ***stratify_labels*** method to help make sure that the train and test dataset are split uniformly based on the candidate labels.

```
from sklearn.model_selection import train_test_split

tag_labels = test_french_df[candidate_labels]
train, test = train_test_split(test_french_df, random_state=42,
test_size=0.20, shuffle=True, stratify=tag_labels)

from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer(strip_accents='unicode', analyzer='word',
ngram_range=(1,3), norm='l2')
vectorizer.fit(train['training_text'])
vectorizer.fit(test['training_text'])

x_train = vectorizer.transform(train['training_text'])
y_train = train.drop(labels = ['Score', 'Label', 'training_text'],
axis=1)

x_test = vectorizer.transform(test['training_text'])
y_test = test.drop(labels = ['Score', 'Label', 'training_text'],
axis=1)
```

2.4.1.2 Decision Tree Classifier French

Decision Tree is a Supervised Machine Learning Algorithm that uses a set of rules to make decisions, similarly to how humans make decisions.

One way to think of a Machine Learning classification algorithm is that it is built to make decisions.

You usually say the model predicts the class of the new, never-seen-before input but, behind the scenes, the algorithm has to decide which class to assign.

We see a reasonable accuracy and results based on the limitations in th dataset and the labels we are using

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
import seaborn as sns

dtc = DecisionTreeClassifier()
dtc.fit(x_train, y_train)
predictions = dtc.predict(x_test)



print("Accuracy = ",accuracy_score(y_test, predictions))


predictions = pd.DataFrame(predictions)
predictions = predictions.rename(columns={0:'anxious', 1:'relief',
2:'sad'})

from sklearn.metrics import classification_report
print('\nClassification Report\n')
print(classification_report(y_test.idxmax(axis=1),
predictions.idxmax(axis=1), target_names=candidate_labels))
```

```
Accuracy =  0.5266666666666666

Classification Report

              precision    recall  f1-score   support

         sad       0.53      0.85      0.65       594
      relief       0.55      0.27      0.37       374
     anxious       0.38      0.12      0.18       232

    accuracy                           0.53      1200
   macro avg       0.49      0.41      0.40      1200
weighted avg       0.51      0.53      0.47      1200
```

2.4.1.3 Random Forest Classifier

Usually, we only have a vague idea of the best hyperparameters and thus the best approach to narrow our search is to evaluate a wide range of values for each hyperparameter. Using Scikit-Learn's ***GridSearchCV*** method, we can define a grid of hyperparameter ranges, and sample from the grid, performing K-Fold CV with each combination of values.

In the end we get a list of optimal hyperparameters for the Random Forest Classifier.

```
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import GridSearchCV

rfc = RandomForestClassifier()
grid_params = {
    'n_estimators' : [90,120],
    'criterion' : ['gini','entropy'],
    'max_depth' : [2, 5, 10],
    'min_samples_leaf' : [1, 5, 10],
    'min_samples_split' : [2, 5, 10],
    'max_features' : ['auto', 'log2']
}

grid_search = GridSearchCV(estimator=rfc, param_grid=grid_params,
cv=5, n_jobs=-1, verbose=3)

grid_search.fit(x_train, y_train)

grid_search.best_params_
```

```
Fitting 5 folds for each of 216 candidates, totalling 1080 fits
```

```
{'criterion': 'gini',
 'max_depth': 10,
 'max_features': 'auto',
 'min_samples_leaf': 1,
 'min_samples_split': 5,
 'n_estimators': 120}
```

Use the optimal hyperparameters as mentioned above for our Random Forest Classifier

```
rfc = RandomForestClassifier(criterion=
grid_search.best_params_['criterion'],
 max_depth = grid_search.best_params_['max_depth'],
 max_features= grid_search.best_params_['max_features'],
 min_samples_leaf = grid_search.best_params_['min_samples_leaf'],
 min_samples_split = grid_search.best_params_['min_samples_split'],
 n_estimators = grid_search.best_params_['n_estimators'])
rfc.fit(x_train, y_train)
predictions = rfc.predict(x_test)

print("Accuracy = ",accuracy_score(y_test, predictions))
```

```
Accuracy =  0.43666666666666665
```

2.4.1.4 One vs Rest Classifier

The Multi-label algorithm accepts a binary mask over multiple labels. The result for each prediction will be an array of 0s and 1s marking which class labels apply to each row input sample.

Logistic Regression

OneVsRest strategy can be used for multi-label learning, where a classifier is used to predict multiple labels for instance. Logistic Regression supports multi-class, but we are in a multi-label scenario, therefore, we wrap *Logistic Regression in the OneVsRestClassifier.

```
from sklearn.linear_model import LogisticRegression
from sklearn.pipeline import Pipeline
from sklearn.metrics import accuracy_score
from sklearn.multiclass import OneVsRestClassifier
# Using pipeline for applying logistic regression and one vs rest
classifier
LogReg_pipeline = Pipeline([
                ('clf',
OneVsRestClassifier(LogisticRegression(solver='sag'), n_jobs=-1)),
            ])

categories = candidate_labels

for category in categories:
    print('**Processing {} tweets...**'.format(category))

    # Training logistic regression model on train data
    LogReg_pipeline.fit(x_train, y_train[category])

    # calculating test accuracy
    prediction = LogReg_pipeline.predict(x_test)
    print('Test accuracy is
{}'.format(accuracy_score(y_test[category], prediction)))
    print("\n")
```

```
**Processing sad tweets...**
Test accuracy is 0.8066666666666666


**Processing relief tweets...**
Test accuracy is 0.7216666666666667


**Processing anxious tweets...**
Test accuracy is 0.6925
```

2.4.2 Implementing an LSTM to potentially increase French

2.4.2.1 Initialising the French LSTM

Vectorize tweets text, by turning each text into either a sequence of integers or into a vector.

Limit the data set to the top 15,000 words.

Set the max number of words in each tweet at 50. This is most of what we see when it comes to number of words used per tweet

First we shall look at the distribution of the labels.

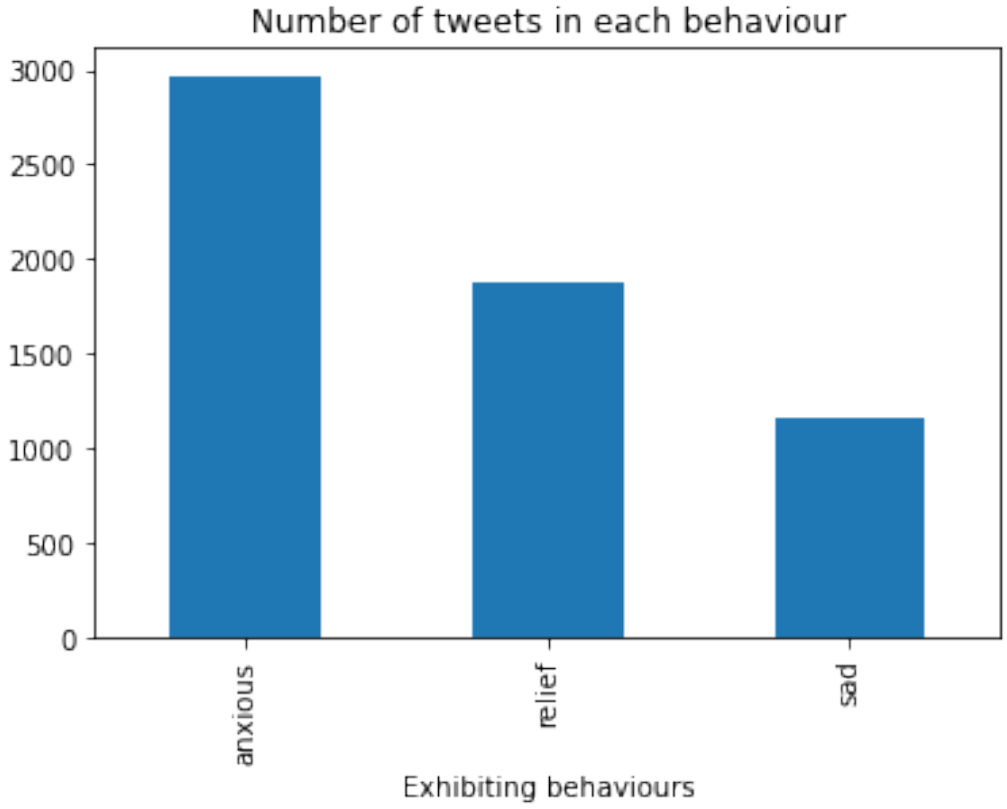
We are using the LSTM to potentially improve the performance of the Classifier on unseen tweets for behavior extraction labels.

```
test_french_df.Label.value_counts()
```

```
anxious    2967
relief     1872
sad        1161
Name: Label, dtype: int64
```

```
test_french_df['Label'].value_counts().sort_values(ascending=False).plot(kind='bar', xlabel='Exhibiting behaviours',

title='Number of tweets in each behaviour')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5718ca4be0>
```



Recurrent Neural Network (RNN) using the Long Short Term Memory (LSTM) architecture can be implemented using Keras.

Reference to Keras library: https://keras.io/

```
from keras_preprocessing.text import Tokenizer
from keras_preprocessing.sequence import pad_sequences
from keras.models import Sequential
from keras.layers import Dense, Embedding, LSTM, SpatialDropout1D
from sklearn.model_selection import train_test_split
from keras.utils.np_utils import to_categorical
from keras.callbacks import EarlyStopping
from keras.layers import Dropout

# The maximum number of words to be used. (most frequent)
MAX_NB_WORDS = 15000
# Max number of words in each tweet.
MAX_SEQUENCE_LENGTH = 50
# This is fixed.
EMBEDDING_DIM = 100

tokenizer = Tokenizer(num_words=MAX_NB_WORDS, filters='!"#$
%&()*+,-./:;<=>?@[\]^_`{|}~', lower=True)
tokenizer.fit_on_texts(test_french_df['training_text'].values)
word_index = tokenizer.word_index
print('Found %s unique tokens.' % len(word_index))
```

```
Found 25561 unique tokens.
```

Truncate and pad the input sequences so that they are all in the same length for modeling.

```
X =
tokenizer.texts_to_sequences(test_french_df['training_text'].values)
X = pad_sequences(X, maxlen=MAX_SEQUENCE_LENGTH)
print('Shape of data tensor:', X.shape)
```

```
Shape of data tensor: (6000, 50)
```

Use the one hot encoded categorical labels as the target

```
Y= test_french_df[candidate_labels]
Y = Y.loc[:,~Y.columns.duplicated()].copy()
print('Shape of label tensor:', Y.shape)
```

```
Shape of label tensor: (6000, 3)
```

Train and Test Split (80/20 respectively)

```
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size =
0.20, random_state = 42)
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(4800, 50) (4800, 3)
(1200, 50) (1200, 3)
```

The first layer is the embedded layer that uses 100 length vectors to represent each word.

SpatialDropout1D performs variational dropout in NLP models.

The next layer is the LSTM layer with 100 memory units.

The output layer must create 3 output values, one for each class.

Activation function is softmax for multi-class classification.

Because it is a multi-class classification problem, categorical_crossentropy is used as the loss function.

```
model = Sequential()
model.add(Embedding(MAX_NB_WORDS, EMBEDDING_DIM,
input_length=X.shape[1]))
model.add(SpatialDropout1D(0.2))
model.add(LSTM(100, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(3, activation='softmax'))
model.compile(loss='categorical_crossentropy', optimizer='adam',
metrics=['accuracy'])
print(model.summary())
```

```
WARNING:tensorflow:Layer lstm_2 will not use cuDNN kernels since it
doesn't meet the criteria. It will use a generic GPU kernel as
fallback when running on GPU.

Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_2 (Embedding)     (None, 50, 100)           1500000

 spatial_dropout1d_2 (Spatia  (None, 50, 100)          0
 lDropout1D)

 lstm_2 (LSTM)               (None, 100)               80400

 dense_2 (Dense)             (None, 3)                 303

=================================================================
Total params: 1,580,703
Trainable params: 1,580,703
Non-trainable params: 0
_________________________________________________________________
None
```

We reduced the number of epochs to help improve accuracy and to reduce overfitting. We shall see the plot referenced in the futre

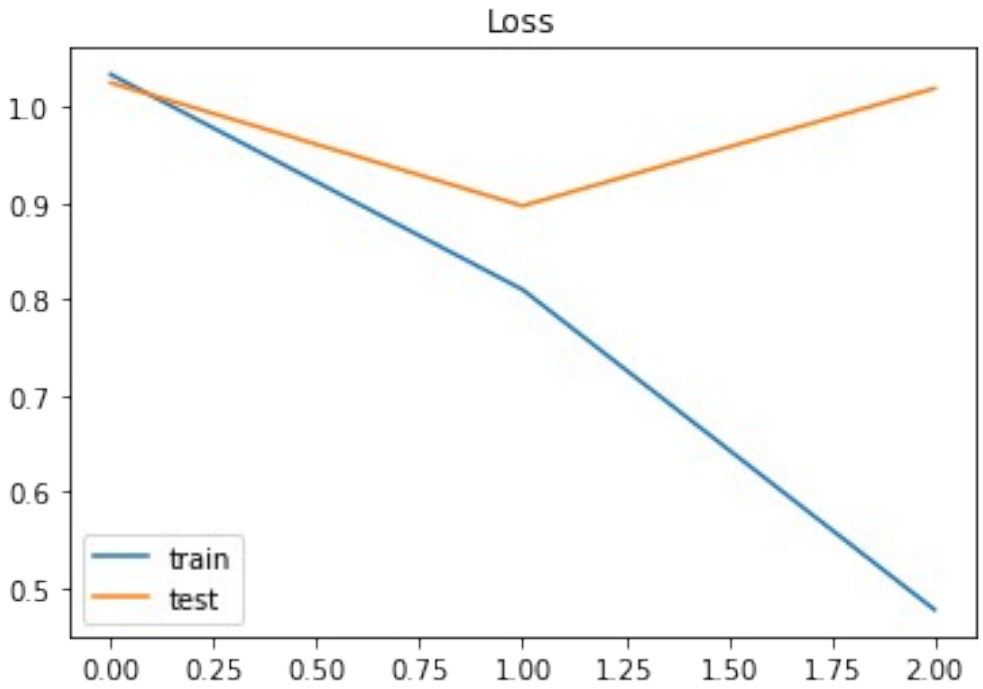
```
epochs = 3
batch_size = 64

history = model.fit(X_train, Y_train, epochs=epochs,
batch_size=batch_size,validation_split=0.1,callbacks=[EarlyStopping(mo
nitor='val_loss', patience=3, min_delta=0.0001)])
```

```
Epoch 1/3
68/68 [==============================] - 25s 331ms/step - loss: 1.0343
- accuracy: 0.4924 - val_loss: 1.0255 - val_accuracy: 0.4812
Epoch 2/3
68/68 [==============================] - 17s 251ms/step - loss: 0.8103
- accuracy: 0.6333 - val_loss: 0.8974 - val_accuracy: 0.5896
Epoch 3/3
68/68 [==============================] - 16s 228ms/step - loss: 0.4770
- accuracy: 0.8199 - val_loss: 1.0197 - val_accuracy: 0.5896
```

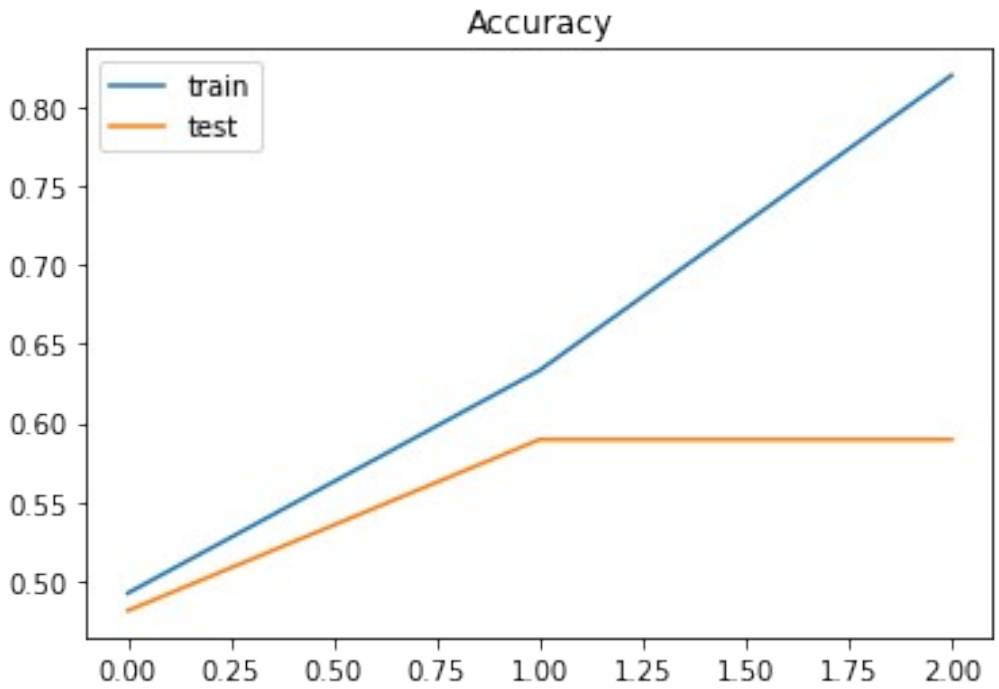
```
accr = model.evaluate(X_test,Y_test)
print('Test set\n  Loss: {:0.3f}\n  Accuracy:
{:0.3f}'.format(accr[0],accr[1]))
```

```
38/38 [==============================] - 1s 18ms/step - loss: 0.9878 -
accuracy: 0.6158
Test set
  Loss: 0.988
  Accuracy: 0.616
```

```
plt.title('Loss')
plt.plot(history.history['loss'], label='train')
plt.plot(history.history['val_loss'], label='test')
plt.legend()
plt.show();
```




```
plt.title('Accuracy')
plt.plot(history.history['accuracy'], label='train')
plt.plot(history.history['val_accuracy'], label='test')
plt.legend()
plt.show();
```



The plots suggest that the model has a little over fitting problem, more data may help, but more epochs will not help using the current data.

2.4.2.2 Inferences based on example tweets French

The tweets used show that the model can give good performance in pulling out behaviours. The test and training data may not be optimal but considering the groun truths used and the overall complexity of NLP modelling the model has great performance on Unseen tweets

Example for tweet showing anxious behavior

```
new_tweet = ["Je suis tellement inquiet de l'état actuel des
répurcussions du vaccin covid, je ne sais pas si je dois prendre le
vaccin ou non"]
seq = tokenizer.texts_to_sequences(new_tweet)
padded = pad_sequences(seq, maxlen=MAX_SEQUENCE_LENGTH)
pred = model.predict(padded)
labels = candidate_labels
print(pred, 'Label predicted by model is:',labels[np.argmax(pred)])
```

```
1/1 [==============================] - 0s 244ms/step
[[0.30303168 0.32467958 0.3722887 ]] Label predicted by model is:
anxious
```

Example for tweet showing relieved behavior

```
new_tweet = ["Les fabricants de vaccins disent que le fait de
continuer à se concentrer sur la préparation du vaccin me soulage"]
seq = tokenizer.texts_to_sequences(new_tweet)
padded = pad_sequences(seq, maxlen=MAX_SEQUENCE_LENGTH)
pred = model.predict(padded)
labels = candidate_labels
print(pred, 'Label predicted by model is:',labels[np.argmax(pred)])
```

```
1/1 [==============================] - 0s 28ms/step
[[0.18939853 0.57198066 0.23862077]] Label predicted by model is:
relief
```

Example for tweet showing sad behavior

```
new_tweet = ["je me sens si triste"]
seq = tokenizer.texts_to_sequences(new_tweet)
padded = pad_sequences(seq, maxlen=MAX_SEQUENCE_LENGTH)
pred = model.predict(padded)
labels = candidate_labels
print(pred, 'Label predicted by model is:',labels[np.argmax(pred)])
```

```
1/1 [==============================] - 0s 29ms/step
[[0.5036491  0.3204508  0.17590007]] Label predicted by model is: sad
```

2.5 Multi-Label Spanish Classification

Due to the limitations of the GPU, we are unable to currently implement the Spanish model into the notebook as the GPU RAM explodes when running ZeroShot learning multiple times.

But as showcased above, we have the capability to run the model on multiple languages and hence have multi-lingual capabilities for the behavior recognition

Part III: Unexpected Challenges and Obstacles

There were a number of unexpected challenges and obstacles that faced us as a group throughout the project. For example, there was a lot of difficulty in trying to find adequate libraries and toolkits to use for multilingual sentiment analysis. This makes sense, as the field is still actively growing and in its early stages, especially in languages other than English. However, finding the other libraries was either immensely difficult or not-lexicon based (aka not using compound negative/positive/neutral sentiment scores), so we had to reduce our planned multilingual analysis from having over 5 languages to just three: English, Spanish, and French.

Furthermore, another large unexpected obstacle that faced us when trying to run our final project was the sheer runtime that was required to run everything, like for the user-defined functions to get the sentiment scores of the processed tweets or RidgePlots, which we originally had. These would often take hours to run, which led us to try and either find solutions that would speed-up the code or just remove that piece of code entirely and replace it with something else. Within our code, these solutions manifested through things like the Numba library and its just-in-time (JIT) compiler to make the user-defined functions faster, or through removal, as seen with the RidgePlots that were replaced with KDE plots instead.

Another side-effect of us having such a large runtime was that we were unable to analyze the entire dataset given, as in total there were over 8 million rows of data when taking all days together. This limited our ability to analyze sentiment at a more comprehensive level, such as across dates. For example, in our analysis we kept the column specifying the date posted despite not using it -- if we had been able to analyze on all days, then we could have used the date column to explore sentiment change over time according to different language tweets.

The problems we faced during modelling were to do with limitations of the GPU and unable to get enough data of tweets with ground labels to work with. A better GPU and more RAM and we would be able to process potentially 50,000 tweets of different sentiments and this would enable us to have more potential behaviors which would help up to predict depression or negative mental health factors form tweets.

Using potential solution such as upsampling or downsampling the data caused an opposite effect and resulted in the model performing worse. This was an interesting result because the model was performing opposite to our intuition on the same. We then decided to use a smaller set of potential behaviors to classify landing on 3 as the optimal for now but this can be enhance with better datasets and GPU performance.

Due to the GPU limitations inherent within Colab we were unable to implement for Spanish in the same way as the French model. However, we have showcased the capabilities of the Multi-lingual Text Classification through the French model and the same could be applied for Spanish

and other languages. Thus, we have shown modeling that can help predict sentiment and indications of depression in English, French, and presumably, given more GPU and RAM, Spanish as well.

Part IV: Conclusions, Future Directions, and Reflection

Conclusions:

We ran this pre-processing, data analysis, and modeling on COVID-19 related tweets during the beginning of early April. Taking the Kaggle dataset, we managed to clean the dataset and split it into preprocessed dataframes for the languages English, French, and Spanish. Then, using sentiment analysis toolkits were reputable and had a history of being used in studies, we were able to get sentiment per tweet.

Then, for modeling, using an LSTM and ML methods such as Decision Tree Classifiers and Random Forests we are able to get reasonable accuracy of the amount of 60% for English and French for detecting potential depression behaviours using the tweets as inputs. Furthermore, as the Spanish tweets were processed in a similar model, and the modeling would be essentially the same, we believe we can find reasonable accuracy for Spanish tweets as well with more computing resources.

Our goal for this model was to be able to use tweets to predict depression exhibiting sentiment so that people who are suffering from Depression or Negative Mental Health could be identified and would be easier to reach out to them using targeted ads and potentially help people to address their situation. We want to able to attack this problem at the root cause and potentially look at social media traffic to identify people showing signs of negative mental health, so this model showing accuracy of roughly 60% is a good start for future development.

Future Directions:

We would like to refine the model further on a good chunk of data and help to increase accuracy based on a wide variety of unseen tweets. The ZeroShot model is trained for multiple languages so we could potentially stack classifiers for behaviors of different languages and create a multi-lingual model which can handle tweets in any language and then could increase and broaden our horizon for analysis. Additionally, we'd like to scale the model to work with a larger quantity of tweets such that we could use data of different demographics, analyze their sentiment, and identify groups that need support.

We would also like to potentially work with data from other Social Media such as Facebook and Instagram to broaden the spectrum of analysis and make our model more robust and universally accepted.

Specifically regarding performance, as currently the runtime takes an extensive amount of time, a future direction that we would like to work towards is getting GPU offloading using cuDF and Rapids to work. We had tested its implementation extensively, but because it didn't seem to be working, we unfortunately had to remove it. Getting this aspect to work would help with runtime massively, however, and by extension allow us to use larger chunks of data.

Likewise, we would like to confirm that this model can predict Spanish depressive sentiment as well, although we assume that it can based off of similar performance by the English and French models. With more GPU space and RAM, we could likely do this.

Finally, if we have many more resources and GPU offloading, then we can try to analyze this on tweets from late April as well, which is in a separate Kaggle dataset by the same author (https://www.kaggle.com/datasets/smid80/coronavirus-covid19-tweets-late-april). This would allow us to have even more data over an even longer stretch of time, which would help us analyze and predict depressive sentiment from earlier in the pandemic compared to later in the pandemic as well.

Reflections:

Computing Multi-Label Text Classification is a challenging endeavour, but considering the limitations we were working with we believe that the model does a reasonable job of classification into the behaviors which would help us to identify individuals showcasing sadness, anxiety or relief.

Working on such a sensitive topic made us realise that it is imperative that we keep working on helping people all over the world in our own way by assisting in identifying people who are potentially at risk. If this work is able to help NGOs or other Social Organisations to reach out to people in need, it would give our work here a purpose and motivation to work further.