```python
In [1]: import pandas as pd
        import matplotlib.pyplot as plt
        plt.style.use('ggplot')
        import seaborn as sns
        import numpy as np
```

```python
In [2]: import os
        os.environ['KAGGLE_USERNAME'] = "shenshu1992" # username from the json file
        os.environ['KAGGLE_KEY'] = "9d596fec3e66e052f965e55754f9524b" # key from the json file
```

```python
In [3]: !kaggle datasets download -d manchunhui/us-election-2020-tweets
```

```
us-election-2020-tweets.zip: Skipping, found more recently modified local copy (use --force to force download)
```

Download datasets from kaggle

```python
In [4]: !unzip us-election-2020-tweets.zip
```

```
Archive:  us-election-2020-tweets.zip
replace hashtag_donaldtrump.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename: A
  inflating: hashtag_donaldtrump.csv
  inflating: hashtag_joebiden.csv
```

unzip the zipfile just download

```python
In [5]: df_j=pd.read_csv('hashtag_joebiden.csv',lineterminator='\n')
```

put csv file in data frame

```python
In [6]: df_j.sample(5)
```

Out[6]:

	created_at	tweet_id	tweet	likes	retweet_count	source	user_id	user_name	user_screen_name	use
501525	2020-11-06 00:58:53	1.324517e+18	#Biden coming thru in the polls right now! Yas...	1.0	0.0	Twitter for iPhone	1.899647e+08	▲keemRakim□	ZamnRakim	▲keem alwa
523121	2020-11-06 13:20:26	1.324703e+18	The (s)Election Chaos Circus is a Nice Break F...	75.0	17.0	Twitter Web App	1.991527e+08	The Dollar Vigilante	DollarVigilante	The D with
701918	2020-11-08 00:17:37	1.325231e+18	Debuted with 62% Spotify Stream from the entir...	0.0	0.0	Twitter for Android	3.253185e+09	These Are Special Times-CelineDion	CelineSuperG	https://t.co
598981	2020-11-07 16:55:12	1.325120e+18	Liebes @ZDF ,\nIn Ihrem Länderspiegel berichte...	1.0	0.0	Twitter for Android	1.260646e+18	Siegfried Goldstein	Siegfried_GSt	Mann Kin
381478	2020-11-04 06:29:18	1.323875e+18	Trump secures another state, now it's Texas\nA...	2.0	1.0	Twitter for Android	1.267470e+18	Adrián Kore Kore Sports	adrian_kore	f d \nfotógra

```python
In [7]: df_j["target"]='joebiden'
```

add a column name "target"

```python
In [8]: df_j = df_j.drop(columns = ['created_at', 'tweet_id',
               'user_id', 'user_name', 'user_screen_name',
               'user_description', 'user_join_date', 'collected_at'])
```

drop useless columns

```python
In [9]: df_j=df_j.rename(columns={"tweet": "text", "who": "target"})
```

```python
In [10]: df_t=pd.read_csv('hashtag_donaldtrump.csv',lineterminator='\n')
```

put csv file in data frame

```python
In [11]: df_t.sample(5)
```

Out[11]:

	created_at	tweet_id	tweet	likes	retweet_count	source	user_id	user_name	user_screen_name
328423	2020-10-29 16:32:33	1.321852e+18	Ma guarda quanti casi strani...\nSa che i Fasc...	1.0	0.0	Twitter for Android	1.282230e+18	Antonella Starace	AntonellaStara3
725436	2020-11-06 00:02:16	1.324502e+18	Russia wants to undermine confidence in our el...	1.0	0.0	Twitter Web App	1.062068e+09	TacosTheCrow	tacosthecrow
588987	2020-11-04 10:04:36	1.323929e+18	Zat er vanmorgen nu iemand van het #CDA bij #W...	3.0	0.0	Twitter for iPhone	1.082735e+18	klaaslokaal@klaaslokaal	klaaslokaal
485997	2020-11-03 13:16:03	1.323615e+18	Let's vote for #Trump #ElectionDay	4.0	1.0	Twitter for iPhone	2.864475e+09	Asaad Gul Awan PK	markhor_2018
70788	2020-10-18 16:31:25	1.317866e+18	Again- a Tяump saying the quiet part out loud!...	27.0	27.0	Twitter for iPhone	9.792121e+17	Save Our Country	BigBlueWaveUSA

```python
In [12]: df_t = df_t.drop(columns = ['created_at', 'tweet_id',
                'user_id', 'user_name', 'user_screen_name',
                'user_description', 'user_join_date', 'collected_at'])
```

drop useless columns

```python
In [13]: df_t["target"]="donaldtrump"
```

add a column name "target"

```python
In [14]: df_t=df_t.rename(columns={"tweet": "text"})
```

rename column "tweet" to "text"

```python
In [15]: df_j_t = pd.concat([df_t, df_j]).reset_index()
```

combine two dataframe together

```python
In [16]: df_j_t = df_j_t.drop(columns = ['index'])
```

drop "index" column

```python
In [17]: df_j_t.head(5)
```

Out[17]:

	text	likes	retweet_count	source	user_followers_count	user_location	lat	long	city	country
0	#Elecciones2020 | En #Florida: #JoeBiden dice ...	0.0	0.0	TweetDeck	1860.0	Philadelphia, PA / Miami, FL	25.774270	-80.193660	NaN	United States of America
1	Usa 2020, Trump contro Facebook e Twitter: cop...	26.0	9.0	Social Mediaset	1067661.0	NaN	NaN	NaN	NaN	NaN
2	#Trump: As a student I used to hear for years,...	2.0	1.0	Twitter Web App	1185.0	Portland	45.520247	-122.674195	Portland	United States of America
3	2 hours since last tweet from #Trump! Maybe he...	0.0	0.0	Trumpytweeter	32.0	NaN	NaN	NaN	NaN	NaN
4	You get a tie! And you get a tie! #Trump 's ra...	4.0	3.0	Twitter for iPhone	5393.0	Washington DC	38.894992	-77.036558	Washington	United States of America

```python
In [18]: df_j_t.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1747805 entries, 0 to 1747804
Data columns (total 14 columns):
 #   Column                Dtype
---  ------                -----
 0   text                  object
 1   likes                 float64
 2   retweet_count         float64
 3   source                object
 4   user_followers_count  float64
 5   user_location         object
 6   lat                   float64
 7   long                  float64
 8   city                  object
 9   country               object
 10  continent             object
 11  state                 object
 12  state_code            object
 13  target                object
dtypes: float64(5), object(9)
memory usage: 186.7+ MB
```

```python
In [19]: df_j_t.head(5)
```

Out[19]:

	text	likes	retweet_count	source	user_followers_count	user_location	lat	long	city	country
0	#Elecciones2020 | En #Florida: #JoeBiden dice ...	0.0	0.0	TweetDeck	1860.0	Philadelphia, PA / Miami, FL	25.774270	-80.193660	NaN	United States of America
1	Usa 2020, Trump contro Facebook e Twitter: cop...	26.0	9.0	Social Mediaset	1067661.0	NaN	NaN	NaN	NaN	NaN
2	#Trump: As a student I used to hear for years,...	2.0	1.0	Twitter Web App	1185.0	Portland	45.520247	-122.674195	Portland	United States of America
3	2 hours since last tweet from #Trump! Maybe he...	0.0	0.0	Trumpytweeter	32.0	NaN	NaN	NaN	NaN	NaN
4	You get a tie! And you get a tie! #Trump 's ra...	4.0	3.0	Twitter for iPhone	5393.0	Washington DC	38.894992	-77.036558	Washington	United States of America

```python
In [20]: df_j_t.describe()
```

Out[20]:

	likes	retweet_count	user_followers_count	lat	long
count	1.747805e+06	1.747805e+06	1.747805e+06	801012.000000	801012.000000
mean	8.670096e+00	1.890890e+00	2.538003e+04	35.434583	-41.083772
std	2.860510e+02	7.101557e+01	3.572733e+05	18.425141	67.666098
min	0.000000e+00	0.000000e+00	0.000000e+00	-90.000000	-175.202642
25%	0.000000e+00	0.000000e+00	7.500000e+01	31.816038	-97.086720
50%	0.000000e+00	0.000000e+00	4.350000e+02	39.783730	-74.006015
75%	2.000000e+00	0.000000e+00	2.072000e+03	45.520247	6.130161
max	1.657020e+05	6.347300e+04	8.241710e+07	90.000000	179.048837

```python
In [21]: df_j_t.columns
```

```
Out[21]: Index(['text', 'likes', 'retweet_count', 'source', 'user_followers_count',
        'user_location', 'lat', 'long', 'city', 'country', 'continent', 'state',
        'state_code', 'target'],
       dtype='object')
```

```python
In [22]: df_text_target = df_j_t.drop(columns =[ 'likes', 'retweet_count', 'source', 'user_followers_count',
        'user_location', 'lat', 'long', 'city', 'country', 'continent', 'state',
        'state_code'] )
```

creat a new dataframe only have text and target

```python
In [23]: df_text_target.head(5)
```

Out[23]:

	text	target
0	#Elecciones2020 | En #Florida: #JoeBiden dice ...	donaldtrump
1	Usa 2020, Trump contro Facebook e Twitter: cop...	donaldtrump
2	#Trump: As a student I used to hear for years,...	donaldtrump
3	2 hours since last tweet from #Trump! Maybe he...	donaldtrump
4	You get a tie! And you get a tie! #Trump 's ra...	donaldtrump

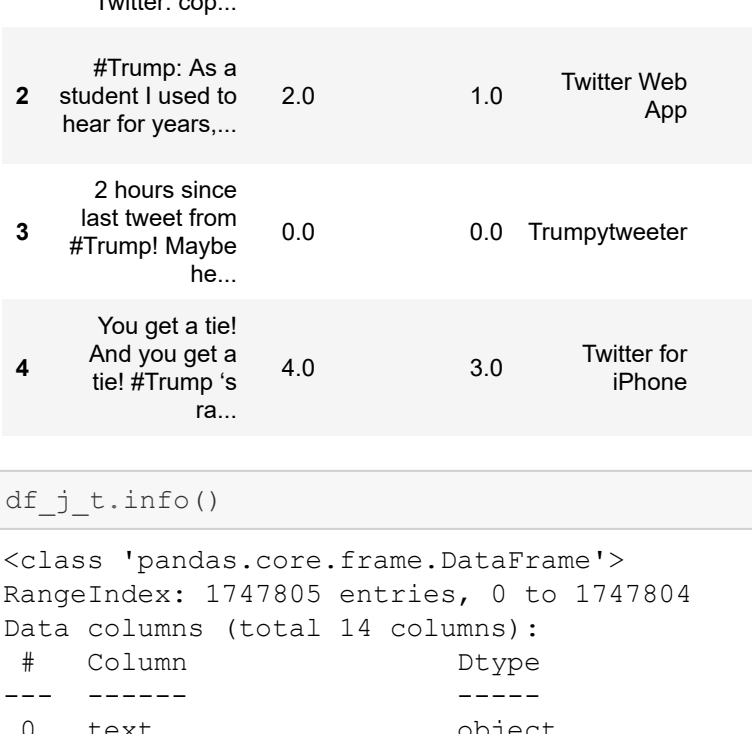
```python
In [24]: df_text_target.count()
```

```
Out[24]: text      1747805
         target    1747805
         dtype: int64
```

```python
In [25]: x = df_text_target["target"].value_counts()
         plt.grid()
         sns.barplot(x.index, x)
         plt.gca().set_ylabel("samples")
         plt.title("distribution")
```

```
/usr/local/lib/python3.6/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variables as keyword args: x, y. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
  FutureWarning
```

```
Out[25]: Text(0.5, 1.0, 'distribution')
```

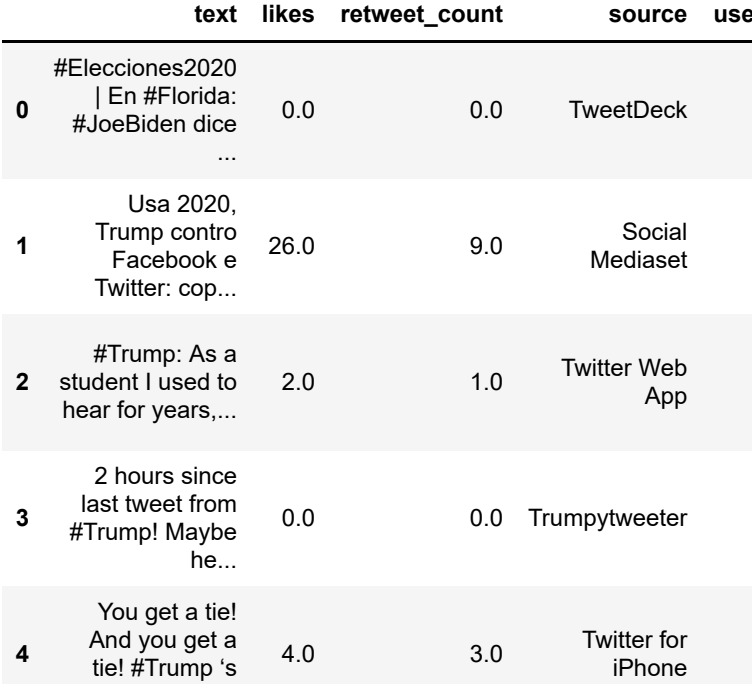



```python
In [26]: plt.grid()
         plt.hist(df_text_target[df_text_target["target"] == "donaldtrump"]["text"].str.len(), color="b")
         plt.title("Trump tweets length")
```

```
Out[26]: Text(0.5, 1.0, 'Trump tweets length')
```

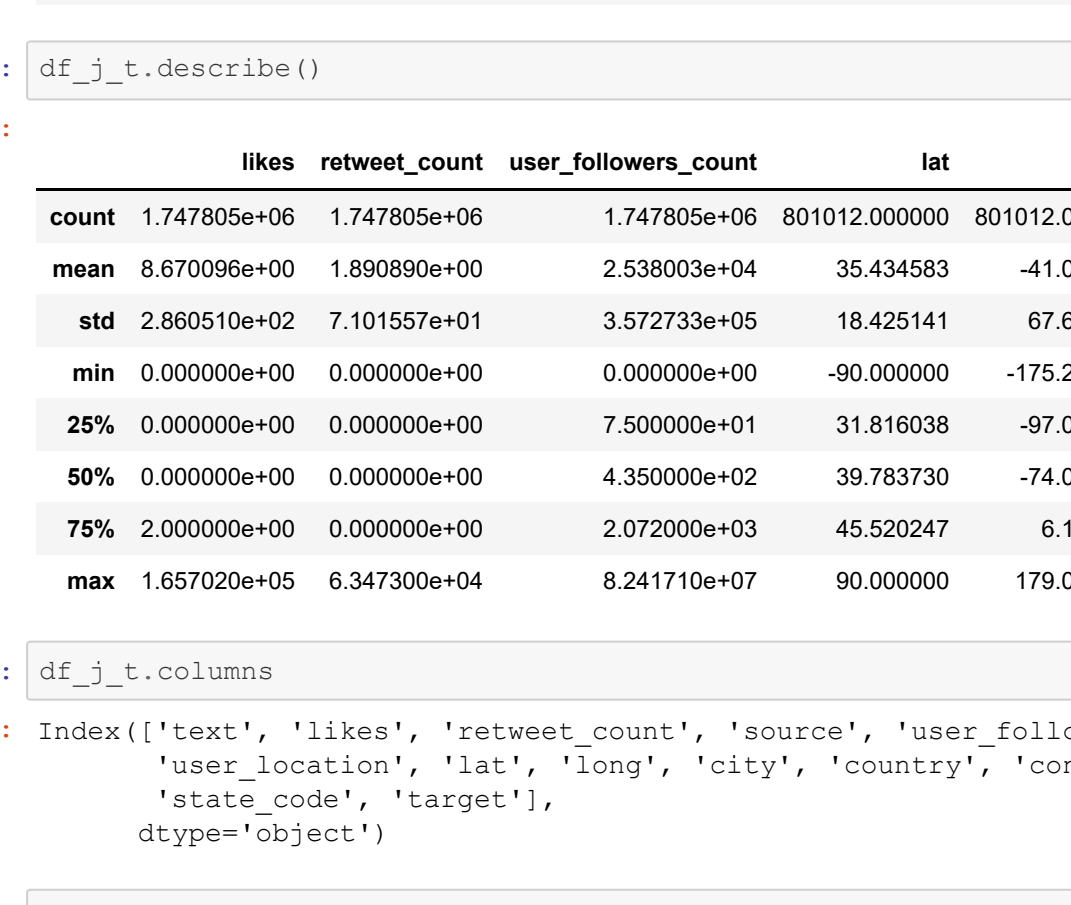

```python
In [27]: plt.grid()
         plt.hist(df_text_target[df_text_target["target"] == "joebiden"]["text"].str.len(),color="r")
         plt.title("Biden tweets length")
```

```
Out[27]: Text(0.5, 1.0, 'Biden tweets length')
```



Biden vs Trump in each continents.

```python
In [28]: sns.catplot(y="continent", data=df_j_t, hue='target',kind='count')
         plt.figure(figsize=(10,5))
         plt.show()
```




```
<Figure size 720x360 with 0 Axes>
```

Most unique words in each category

```python
In [29]: features_categorical=['source',
               'user_location', 'city', 'country','state','target']
         for i in features_categorical:
           print(df_j_t[i].value_counts()[:5])
           print('==========================================================')
```

```
Twitter Web App        561380
Twitter for iPhone     518843
Twitter for Android    488212
Twitter for iPad        61362
TweetDeck               29988
Name: source, dtype: int64
==========================================================
United States      39253
USA                16105
Washington, DC     10831
India              10680
California, USA     9904
Name: user_location, dtype: int64
==========================================================
New York       37749
London         19204
Los Angeles    16838
Washington     16475
Paris          16173
Name: city, dtype: int64
==========================================================
United States of America    332495
United States                61905
United Kingdom               58051
India                        40091
Germany                      35379
Name: country, dtype: int64
==========================================================
California    56966
New York      43886
England       40854
Texas         30682
Florida       29852
Name: state, dtype: int64
==========================================================
donaldtrump    970919
joebiden       776886
Name: target, dtype: int64
==========================================================
```

```python
In [30]: import plotly.graph_objects as go
```

The top 10 sources of tweets

```python
In [31]: y= df_j_t.query('target == "joebiden"').groupby(by='source').count().text.sort_values(ascending=False)[:10]
         x= df_j_t.query('target == "joebiden"').groupby(by='source').count().text.sort_values(ascending=False)[:10].index
         y2= df_j_t.query('target == "donaldtrump"').groupby(by='source').count().text.sort_values(ascending=False)[:10]
         x2= df_j_t.query('target == "donaldtrump"').groupby(by='source').count().text.sort_values(ascending=False)[:10].index
         fig_1 = go.Figure([go.Bar(x=x, y=y, name='joe Biden'),
                       go.Bar(x=x2, y=y2, name='donaldtrump')])
         fig_1.update_layout(title_text='top 10 sources')
         fig_1.update_xaxes(title='sources')
         fig_1.update_yaxes(title='text count')
         fig_1.show(renderer="colab")
```

Tweets count for each state of America

```python
In [32]: y = df_j_t.query('(target == "joebiden") & (country == "United States of America")').groupby(
             by='state').count().text.sort_values(ascending=False)
         x = df_j_t.query('(target == "joebiden") & (country == "United States of America")').groupby(
             by='state').count().text.sort_values(ascending=False).index
         y2 = df_j_t.query('(target == "donaldtrump") & (country == "United States of America")').groupby(
             by='state').count().text.sort_values(ascending=False)
         x2 = df_j_t.query('(target == "donaldtrump") & (country == "United States of America")').groupby(
             by='state').count().text.sort_values(ascending=False).index
         fig_2 = go.Figure([go.Bar(x=x, y=y, name='joebiden'),
                       go.Bar(x=x2, y=y2, name='donald Trump')])
         fig_2.update_layout(title_text='Tweets count for each state of America')
         fig_2.update_xaxes(title='users')
         fig_2.update_yaxes(title='tweets count')
         fig_2.show(renderer="colab")
```

```python
In [33]: import re
```

Text Clean

Remove URL

```python
In [34]: # Function for url's
         import re
         def remove_urls(text):
             url_pattern = re.compile(r'https?://\S+|www\.\S+')
             return url_pattern.sub(r'link', text)
```

```python
In [35]: df_text_target['text'] = df_text_target['text'].apply(remove_urls)
```

Lowercase

```python
In [36]: df_text_target['text'] = df_text_target['text'].str.lower()
         display(df_text_target['text'].head(5))
```

```
0    #elecciones2020 | en #florida: #joebiden dice ...
1    usa 2020, trump contro facebook e twitter: cop...
2    #trump: as a student i used to hear for years,...
3    2 hours since last tweet from #trump! maybe he...
4    you get a tie! and you get a tie! #trump 's ra...
Name: text, dtype: object
```

```python
In [37]: not_list = ["don", "ain", "ain't", "aren", "arent", "aren't", "cannot", "cant", "can't", "couldn", "couldnt", "couldn't", "didn",
                     "didn't", "doesn", "doesn't", "don", "don't", "hadn", "hadn't", "hasn", "hasnt", "hasn't", "haven", "haven't", "mightn", "mightn't",
                     "isn", "isn't",  "mustn", "mustn't", "needn", "needn't", "nt", "shouldn", "shouldn't", "wasn", "wasnt", "wasn't", "don't"]
```

```python
In [38]: def before_lowercase(text):
             text =  re.sub(r"|".join(not_list), "not", text)
             return str(text)
```

```python
In [39]: df_text_target['text'] = df_text_target['text'].apply(before_lowercase)
         df_text_target['text'].head(6)
```

```
Out[39]: 0    #elecciones2020 | en #florida: #joebiden dice ...
         1    usa 2020, trump conotro facebook e twitter: co...
         2    #trump: as a studenot i used to hear for years...
         3    2 hours since last tweet from #trump! maybe he...
         4    you get a tie! and you get a tie! #trump 's ra...
         5    @clady62 her 15 minutes were over long time ag...
         Name: text, dtype: object
```

remove Emojis&Emoticons

```python
In [40]: def remove_emoji(text):
             emoji_pattern = re.compile("["
                                    u"\U0001F600-\U0001F64F"  # emoticons
                                    u"\U0001F300-\U0001F5FF"  # symbols & pictographs
                                    u"\U0001F680-\U0001F6FF"  # transport & map symbols
                                    u"\U0001F1E0-\U0001F1FF"  # flags (iOS)
                                    u"\U00002702-\U000027B0"
                                    u"\U000024C2-\U0001F251"
                                    "]+", flags=re.UNICODE)
             return emoji_pattern.sub(r'', text)
```

```python
In [41]: df_text_target['text'] = df_text_target['text'].apply(lambda x:remove_emoji(x))
```

```python
In [42]: df_text_target.head(5)
```

Out[42]:

	text	target
0	#elecciones2020 | en #florida: #joebiden dice ...	donaldtrump
1	usa 2020, trump conotro facebook e twitter: co...	donaldtrump
2	#trump: as a studenot i used to hear for years...	donaldtrump
3	2 hours since last tweet from #trump! maybe he...	donaldtrump
4	you get a tie! and you get a tie! #trump 's ra...	donaldtrump

```python
In [43]: from nltk.corpus import stopwords
```

remove stopwoards

```python
In [44]: import nltk
         nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
Out[44]: True
```

```python
In [45]: my_stopwords=set(stopwords.words("english"))
```

```python
In [46]: print(len(my_stopwords))
```

```
179
```

```python
In [47]: def remove_sw(text,s_list):
             a=[]
             for s in text.split():
                 if s not in my_stopwords:
                     a.append(s)
             return a
```

```python
In [48]: b=[]
         for t in df_text_target['text']:
             b.append(remove_sw(t ,my_stopwords))
```

```python
In [49]: df_text_target['text2'] = b
         df_text_target['text2'].head(5)
```

```
Out[49]: 0    [#elecciones2020, |, en, #florida:, #joebiden,...
         1    [usa, 2020,, trump, conotro, facebook, e, twit...
         2    [#trump:, studenot, used, hear, years,, ten, y...
         3    [2, hours, since, last, tweet, #trump!, maybe,...
         4    [get, tie!, get, tie!, #trump, 's, rally, #iow...
         Name: text2, dtype: object
```

```python
In [50]: #Combine individual words
         def combine_text(input):
             combined = ' '.join(input)
             return combined
         df_text_target['text'] = df_text_target['text2'].apply(combine_text)
         df_text_target['text']
```

```
Out[50]: 0          #elecciones2020 | en #florida: #joebiden dice ...
         1          usa 2020, trump conotro facebook e twitter: co...
         2          #trump: studenot used hear years, ten years, h...
         3          2 hours since last tweet #trump! maybe busy. t...
         4                get tie! get tie! #trump 's rally #iowa link
                                          ...
         1747800    stop laying @cnn ! #paris #lonnot nott give fu...
         1747801    ωχ ελπίζω να μη μας βγει σαν τους οπαδούς του ...
         1747802    l'otan va sortir de sa léthargie et redevenir ...
         1747803    "#congiunotifuoriregione" 'sono felice per #ji...
         1747804    ik moet zeggen dat ik #biden "the lesser two e...
         Name: text, Length: 1747805, dtype: object
```

Remove punctuation

```python
In [51]: import string
```

```python
In [52]: def remove_punctuation(text):
             table = str.maketrans('', '', string.punctuation)
             return text.translate(table)
```

```python
In [53]: df_text_target["text"] = df_text_target["text"].apply(lambda x: remove_punctuation(x))
```

```python
In [54]: df_text_target["text"]
```

```
Out[54]: 0          elecciones2020  en florida joebiden dice que n...
         1          usa 2020 trump conotro facebook e twitter copr...
         2          trump studenot used hear years ten years heard...
         3          2 hours since last tweet trump maybe busy trem...
         4                   get tie get tie trump 's rally iowa link
                                          ...
         1747800    stop laying cnn  paris lonnot nott give fuck b...
         1747801    ωχ ελπίζω να μη μας βγει σαν τους οπαδούς του ...
         1747802    lotan va sortir de sa léthargie et redevenir l...
         1747803    "congiunotifuoriregione" 'sono felice per jill...
         1747804    ik moet zeggen dat ik biden the lesser two evi...
         Name: text, Length: 1747805, dtype: object
```

Sentiment Analysis - what's the attitude of tweeter users toward Trump and Biden?

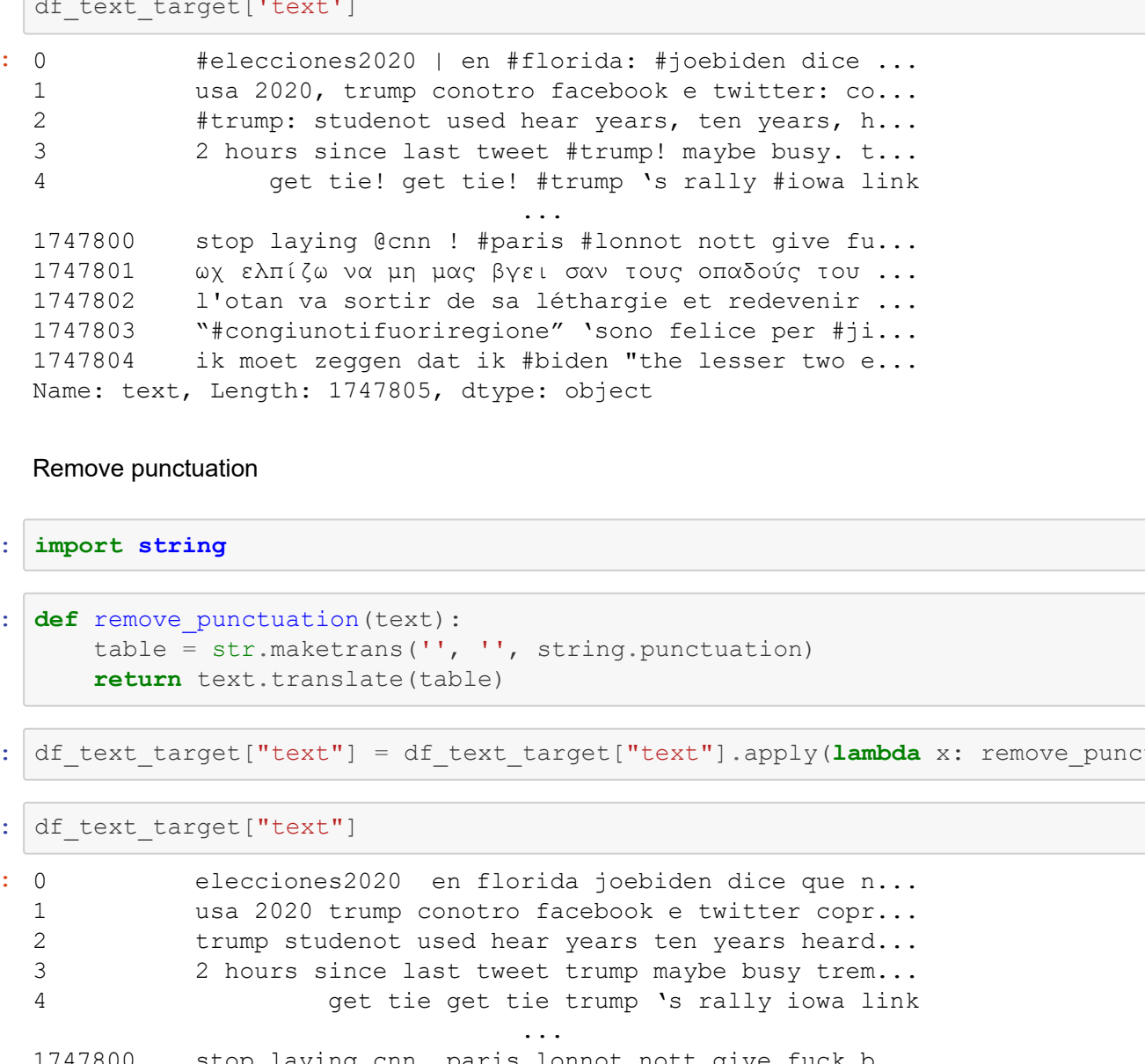
Explore the Data

```python
In [55]: temp = df_text_target.groupby('target').count()['text'].reset_index().sort_values(by='text',ascending=False)
         temp
```

Out[55]:

	target	text
0	donaldtrump	970919
1	joebiden	776886

```python
In [56]: plt.figure(figsize=(9,6))
         sns.countplot(y=df_j_t.country, order = df_j_t.country.value_counts().iloc[:20].index)
         plt.title('Top 20 locations')
         plt.show()
```



Most Common Words

```python
In [57]: from collections import Counter
         cnt = Counter()
         for text in df_text_target["text"].values:
             for word in text.split():
                 cnt[word] += 1

         cnt.most_common(10)
```

```
Out[57]: [('trump', 1232839),
          ('link', 1134227),
          ('biden', 818980),
          ('joebiden', 430727),
          ('de', 220338),
          ('election2020', 181076),
          ('notaldtrump', 173666),
          ('vote', 163702),
          ('realnotaldtrump', 140039),
          ('la', 130200)]
```

```python
In [58]: df_text_target["target"]=df_text_target["target"].map({"joebiden":0 ,"donaldtrump":1})
```

Analysis for Trump

```python
In [59]: def clean(text):
           #Make text lowercase, remove text in square brackets,remove links,remove punctuation and remove words containing numbers
             text = str(text).lower()
             text = re.sub('\[.*?\]', '', text)
             text = re.sub('https?://\S+|www\.\S+', '', text)
             text = re.sub('<.*?>+', '', text)
             text = re.sub('\n', '', text)
             text = re.sub('\w*\d\w*', '', text)
             text=re.sub(r'@[A-Za-z0-9]+','',text)
             text=re.sub(r'#','',text)
             text=re.sub(r'RT[\s]+','',text)
             text=re.sub(r'[^\w]', ' ', text)
             return text
```

```python
In [60]: def getSubjectivity(text):
             return TextBlob(text).sentiment.subjectivity
         def getPolarity(text):
             return TextBlob(text).sentiment.polarity
         def getAnalysis(score):
             if score < 0:
                 return 'negative'
             elif score==0:
                 return 'neutral'
             else:
                 return 'positive'
```

```python
In [61]: Trump_Tweets = df_j_t.query('(target == "donaldtrump")').sort_values('user_followers_count',ascending = False)
```

```python
In [62]: Trump_Tweets = Trump_Tweets.dropna().loc[Trump_Tweets.country == 'United States of America']
         Trump_Tweets.reset_index(inplace = True, drop = True)
```

```python
In [63]: Trump_Tweets['ClearTweet'] = Trump_Tweets['text'].apply(clean)
```

```python
In [64]: from textblob import TextBlob
```

```python
In [65]: Trump_Tweets['subjectivity']= Trump_Tweets['ClearTweet'].apply(getSubjectivity)
         Trump_Tweets['polarity']    = Trump_Tweets['ClearTweet'].apply(getPolarity)
         Trump_Tweets['analysis']    = Trump_Tweets['polarity'].apply(getAnalysis)
         Trump_Tweets.head()
```

Out[65]:

	text	likes	retweet_count	source	user_followers_count	user_location	lat	long	city	country	contin
0	Without a doubt, the #FourSeasons debacle is t...	206.0	12.0	Twitter Web App	5747472.0	Los Angeles, CA	34.053691	-118.242766	Los Angeles	United States of America	No Amer
1	Woah. Have you read this article?\n\nAll the p...	695.0	291.0	Twitter for iPhone	3750110.0	Los Angeles	34.053691	-118.242766	Los Angeles	United States of America	No Amer
2	The latest episode of #SNL tackled #DonaldTrum...	149.0	15.0	SocialFlow	3264802.0	Hollywood, CA	34.098003	-118.329523	Los Angeles	United States of America	No Amer
3	#NatalieMorales explains why some Latinx voter...	83.0	10.0	SocialFlow	3264585.0	Hollywood, CA	34.098003	-118.329523	Los Angeles	United States of America	No Amer
4	Actress #NatalieMorales explained why some Lat...	99.0	21.0	SocialFlow	3264524.0	Hollywood, CA	34.098003	-118.329523	Los Angeles	United States of America	No Amer

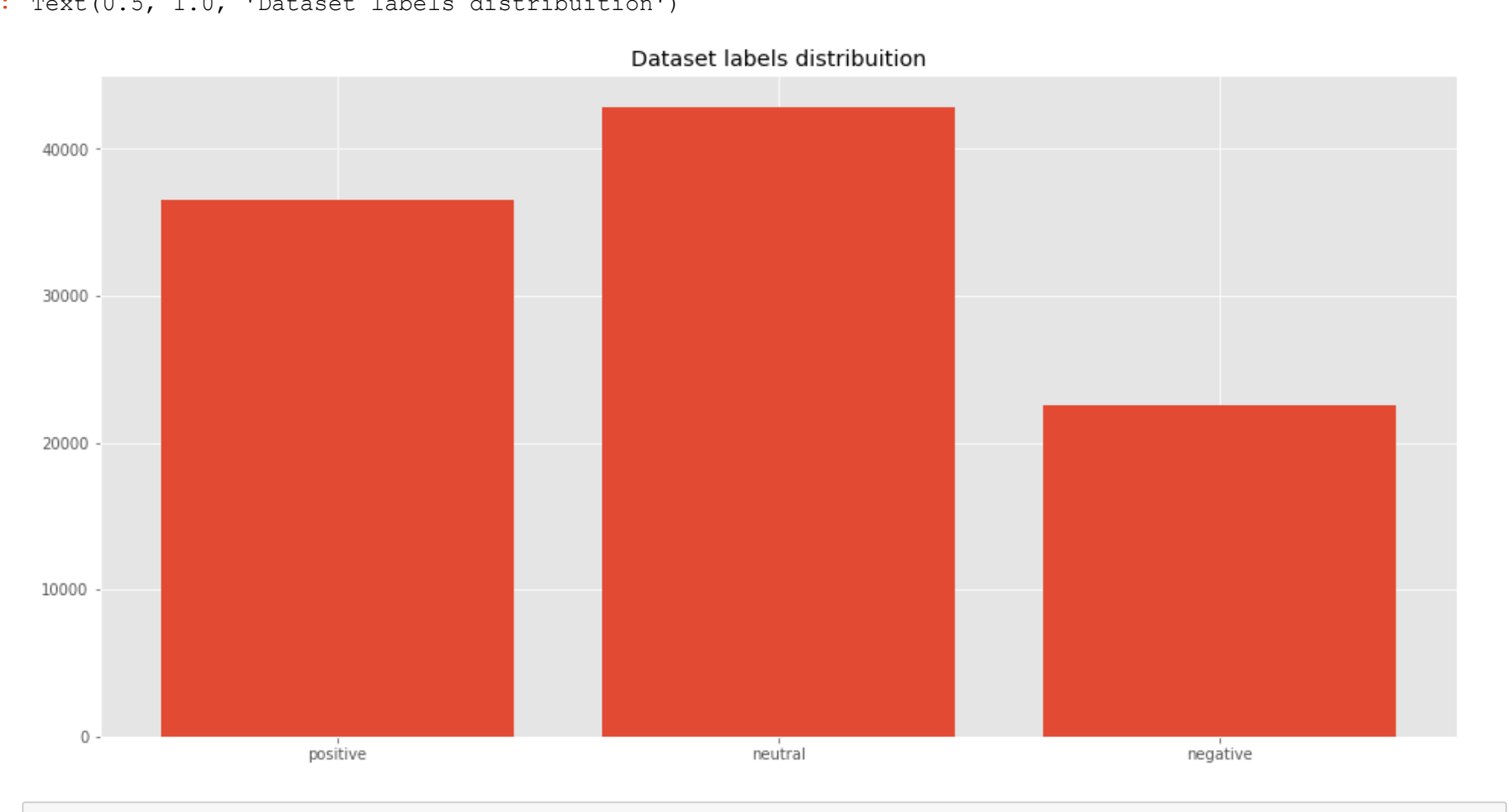
```python
In [66]: Trump_Tweets.polarity = Trump_Tweets.polarity.apply(lambda x: getAnalysis(x))
```

In [67]:
```python
from collections import Counter

target_cnt = Counter(Trump_Tweets.polarity)

plt.figure(figsize=(16,8))
plt.bar(target_cnt.keys(), target_cnt.values())
plt.title("Dataset labels distribuition")
```

Out[67]: Text(0.5, 1.0, 'Dataset labels distribuition')



In [68]:
```python
import plotly.express as px
```

In [69]:
```python
general  = Trump_Tweets.groupby('analysis').analysis.count()
neutral  = Trump_Tweets[Trump_Tweets['analysis'] == 'neutral'].ClearTweet.count()
positive = Trump_Tweets[Trump_Tweets['analysis'] == 'positive'].ClearTweet.count()
negative = Trump_Tweets[Trump_Tweets['analysis'] == 'negative'].ClearTweet.count()

fig = go.Figure(data = [go.Pie(labels = ["positivity","negativity","neutrality"], values = [positive,ne
gative,neutral])])
fig.update_layout(title_text ='Sentiment Analysis: Tweets Donald Trump')
fig.show(renderer ="colab")
```

In [70]:
```python
from wordcloud import WordCloud, STOPWORDS,ImageColorGenerator
```

In [71]:
```python
def word_cloud(wd_list):
    stopwords = set(STOPWORDS)
    all_words = ' '.join([text for text in wd_list])
    wordcloud = WordCloud(
        background_color='white',
        stopwords=stopwords,
        width=1600,
        height=800,
        random_state=1,
        colormap='jet',
        max_words=80,
        max_font_size=200).generate(all_words)
    plt.figure(figsize=(12, 10))
    plt.axis('off')
    plt.imshow(wordcloud, interpolation="bilinear");
word_cloud(df_t['text'][0:1000])
```

Analysis for Biden

In [72]:
```python
Biden_Tweets = df_j_t.query('(target == "joebiden")').sort_values('user_followers_count',ascending = Fa
lse)
```

In [73]:
```python
Biden_Tweets = Biden_Tweets.dropna().loc[Biden_Tweets.country == 'United States of America'][:1000]
Biden_Tweets.reset_index(inplace = True, drop = True)
```

In [74]:
```python
Biden_Tweets['ClearTweet'] = Biden_Tweets['text'].apply(clean)
```

In [75]:
```python
Biden_Tweets['subjectivity']= Biden_Tweets['ClearTweet'].apply(getSubjectivity)
Biden_Tweets['polarity']    = Biden_Tweets['ClearTweet'].apply(getPolarity)
Biden_Tweets['analysis']    = Biden_Tweets['polarity'].apply(getAnalysis)
Biden_Tweets.head()
```

Out[75]:

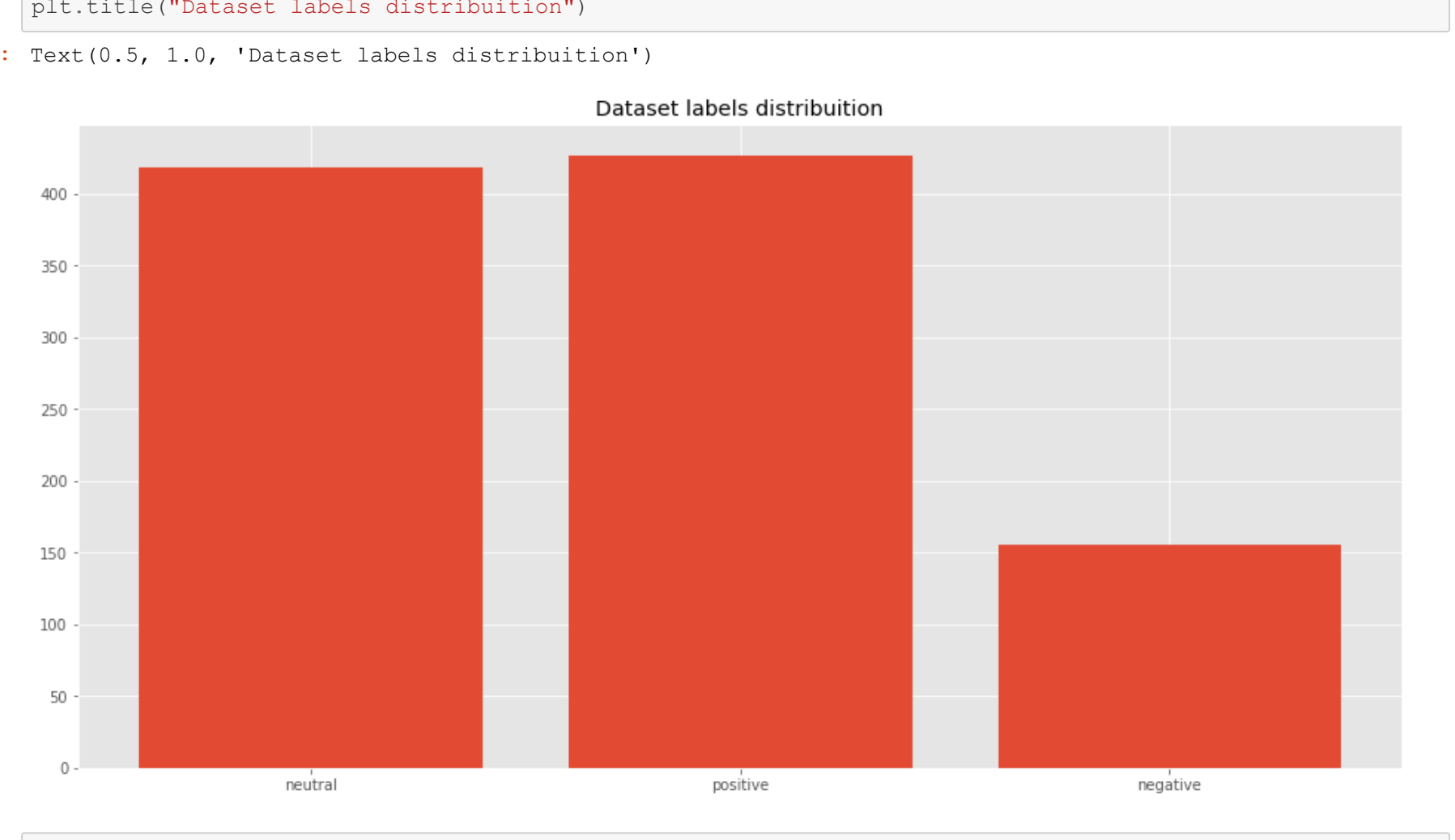
	text	likes	retweet_count	source	user_followers_count	user_location	lat	long	city	country
0	#JoeBiden is enlisting A-list star power to he...	73.0	4.0	Twitter Web App	5750841.0	Los Angeles, CA	34.053691	-118.242766	Los Angeles	United States of America
1	Celebs are just as excited as we are that Amer...	226.0	14.0	Hootsuite Inc.	5747840.0	Los Angeles, CA	34.053691	-118.242766	Los Angeles	United States of America
2	While we were campaigning across battleground ...	717.0	55.0	Twitter Media Studio	5477365.0	Chicago, IL	41.875562	-87.624421	Chicago	United States of America
3	The big fight TODAY!!!!! @realDonaldTrump vs #...	22641.0	2543.0	Twitter for iPhone	5476752.0	Las Vegas, NV	36.167256	-115.148516	Las Vegas	United States of America
4	During his victory speech, #JoeBiden said he u...	406.0	40.0	Twitter for Advertisers (legacy)	3264793.0	Hollywood, CA	34.098003	-118.329523	Los Angeles	United States of America

In [76]:
```python
Biden_Tweets.polarity = Biden_Tweets.polarity.apply(lambda x: getAnalysis(x))
```

In [77]:
```python
target_cnt = Counter(Biden_Tweets.polarity)

plt.figure(figsize=(16,8))
plt.bar(target_cnt.keys(), target_cnt.values())
plt.title("Dataset labels distribuition")
```

Out[77]: Text(0.5, 1.0, 'Dataset labels distribuition')



In [78]:
```python
general  = Biden_Tweets.groupby('analysis').analysis.count()
neutral  = Biden_Tweets[Biden_Tweets['analysis'] == 'neutral'].ClearTweet.count()
positive = Biden_Tweets[Biden_Tweets['analysis'] == 'positive'].ClearTweet.count()
negative = Biden_Tweets[Biden_Tweets['analysis'] == 'negative'].ClearTweet.count()

fig = go.Figure(data = [go.Pie(labels = ["positivity","negativity","neutrality"], values = [positive,ne
gative,neutral])])
fig.update_layout(title_text ='sentimat analysis tweets Joe Biden')
fig.show(renderer = "colab")
```

In [89]:
```python
def word_cloud(wd_list):
    stopwords = set(STOPWORDS)
    all_words = ' '.join([text for text in wd_list])
    wordcloud = WordCloud(
        background_color='white',
        stopwords=stopwords,
        width=1600,
        height=800,
        random_state=1,
        colormap='jet',
        max_words=80,
        max_font_size=200).generate(all_words)
    plt.figure(figsize=(12, 10))
    plt.axis('off')
    plt.imshow(wordcloud, interpolation="bilinear");
word_cloud(df_j['text'][0:1000])
```

change target to binary

In [80]:
```python
df_text_target=df_text_target.drop(columns=["text2"])
```

In [81]:
```python
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
```

In [82]:
```python
X = df_text_target.text
y = df_text_target.target
```

In [83]:
```python
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.1)
```

In [84]:
```python
from sklearn.feature_extraction.text import TfidfVectorizer

tweets = df_text_target['text']
vectorizer = TfidfVectorizer(stop_words='english')

# Learn vocabulary from sentences.
vectorizer.fit(tweets)

# Get vocabularies.
vectorizer.vocabulary_
```

Out[84]:
```
{'elecciones2020': 241073,
 'en': 247675,
 'florida': 281782,
 'joebiden': 389505,
 'dice': 211471,
 'que': 603112,
 'notaldtrump': 523556,
 'solo': 683203,
 'se': 658415,
 'preocupa': 585912,
 'por': 579886,
 'él': 843477,
 'mismo': 486856,
 'el': 240581,
 'demócrata': 203236,
 'fue': 293027,
 'anfitrión': 59030,
 'encuenotros': 248214,
 'electores': 242643,
 'pembrokepines': 561649,
 'miramar': 486153,
 'clic': 158431,
 'aquí': 68005,
 'link': 442128,
 'elsollatino': 245031,
 'yobrilloconelsol': 831758,
 'usa': 776297,
 '2020': 8631,
 'trump': 748721,
 'conotro': 170774,
 'facebook': 266819,
 'twitter': 763800,
 'coprono': 173574,
 'biden': 103767,
 'studenot': 701467,
 'used': 777679,
 'hear': 334464,
 'years': 829633,
 'heard': 334468,
 'china': 151062,
 '2019': 8593,
 '15': 4628,
 'nott': 526427,
 'know': 415070,
 'asked': 73899,
 'said': 645233,
 'sir': 675127,
 'millions': 484331,
 'like': 440468,
 '300': 14509,
 'million': 484250,
 'um': 768780,
 'hours': 346435,
 'tweet': 763023,
 'maybe': 470699,
 'busy': 130518,
 'tremendously': 745883,
 'tie': 734259,
 'rally': 607911,
 'iowa': 371812,
 'clady62': 157082,
 'minutes': 485918,
 'long': 446960,
 'time': 735151,
 'ago': 40533,
 'omarosa': 538804,
 'represenoted': 622783,
 'black': 112396,
 'community': 164916,
 'thereidout': 729769,
 'cried': 181065,
 'begging': 95690,
 'job': 388936,
 'richardmarx': 629199,
 'glad': 309877,
 'got': 315702,
 'house': 346471,
 'dicktrump': 211745,
 'deeviousdenise': 197719,
 'realnotaldtrump': 612952,
 'nypost': 530468,
 'won': 820445,
 'unless': 773098,
 'voting': 800456,
 'agnot': 40259,
 'god': 312003,
 'prevails': 588673,
 'bo': 116593,
 'corrupt': 175359,
 'presidenot': 586715,
 'dark': 191095,
 'light': 440174,
 'lies': 439467,
 'coming': 164071,
 'wouldn': 822452,
 'forever': 284767,
 'single': 674552,
 'effective': 237964,
 'remedies': 620383,
 'eradicate': 253456,
 'round': 637953,
 'plague': 572152,
 'whitehouse': 813960,
 'election2020': 241444,
 'freedomofspeech': 289619,
 'wrong': 823241,
 'cory': 175940,
 'bookers': 119063,
 'brillianot': 124576,
 'final': 277801,
 'questioning': 603863,
 'nominee': 520552,
 'amy': 56389,
 'coney': 167741,
 'barrett': 90339,
 'amyconeybarrett': 56438,
 'corybooker': 175945,
 'booker': 119060,
 'kamalaharris': 401224,
 'scotus': 657462,
 'supremecourtconfirmation': 707021,
 'censorship': 145020,
 'censored': 144982,
 'manipulate': 462001,
 'election': 241425,
 'favor': 272509,
 'agnotst': 40429,
 'ccp': 143339,
 'porn': 580067,
 'fine': 278409,
 'jack': 378877,
 'vijaya': 792232,
 'dickc': 211622,
 'katies': 404927,
 'marciadorsey': 463799,
 'sick': 671780,
 'presidenottrump': 587742,
 'trump2020landslidevictory': 748864,
 'trump2020': 748781,
 'maga': 456848,
 'kag': 400188,
 '4moreyears': 19496,
 'america': 52232,
 'americafirst': 52459,
 'alllivesmatter': 48414,
 'winning': 817411,
 'vote': 797826,
 'voteinperson': 798928,
 'votetrump': 800204,
 'votepresidenottrump': 799329,
 'susanhutch': 707950,
 'ukrnote': 768066,
 'impeachmenot': 358450,
 'factcheck': 267331,
 'amp': 55258,
 'gop': 314755,
 'lied': 439283,
 'tell': 720530,
 'politicians': 577256,
 'stick': 697324,
 'free': 289393,
 'item': 376101,
 'deomocrat': 204134,
 'politician': 577216,
 'politics': 577410,
 'republican': 623163,
 'vpdebate': 801097,
 'freebiedepot': 289453,
 'meiselasb': 474972,
 'wonder': 820486,
 'drugs': 227557,
 'takes': 714008,
 'thats': 724985,
 'masses': 468311,
 'burgers': 129550,
 'hes': 338227,
 'dexamethasone': 209835,
 'high': 339426,
 'currenotly': 185470,
 'november': 527602,
 'latest': 428974,
 'hell': 335984,
 'low': 450229,
 'trumpcovid19': 750329,
 'trumpisalaughingstock': 752662,
 'trumpisaloser': 752680,
 'votehimout': 798864,
 'votebluetoendthnotightmare': 798244,
 'open': 542042,
 'create': 180309,
 'pickem': 569093,
 'pool': 579121,
 'add': 35332,
 'excitemenot': 263121,
 'potenotial': 581943,
 'chaos': 147304,
 'pick': 569075,
 'think': 731729,
 'win': 816844,
 'swing': 709888,
 'states': 694494,
 'popular': 579638,
 'maga2020': 456855,
 'uselection2020': 777773,
 'cnnbrk': 160157,
 'owes': 549102,
 'ricardoaguirre': 628904,
 '730000': 23758,
 'pay': 559018,
 'mass': 468152,
 'murder': 499659,
 'family': 269825,
 'trumpliedpeopledied': 754011,
 'faucifan': 272252,
 'democrats': 201822,
 'spenot': 689110,
 'tax': 717304,
 'payer': 559098,
 'paid': 550987,
 'money': 491892,
 'chasing': 148230,
 'attacking': 77208,
 'doing': 221477,
 'jobs': 389083,
 'serving': 664887,
 'public': 598094,
 'elected': 241309,
 'good': 313938,
 'reasons': 613583,
 'economically': 235811,
 'bail': 86939,
 'people': 562955,
 'protect': 594373,
 'took': 740466,
 'oath': 531466,
 'graves': 318204,
 'away': 82506,
 'rallies': 607831,
 'maskless': 467853,
 'canned': 137328,
 'sardines': 649747,
 'covid19': 178151,
 'superspreader': 706127,
 'trumprally': 755733,
 'trumprallydesmoines': 755751,
 'pandemic': 552384,
 'likes': 440608,
 'farmers': 270987,
 'better': 101527,
 'way': 807309,
 'working': 821402,
 'asses': 74744,
 'check': 148856,
 'totally': 741872,
 'mail': 458327,
 'right': 630732,
 'tedcruz': 719314,
 'ccl25': 143180,
 'blm': 114556,
 'obama': 531525,
 'votered': 799386,
 'bidenharris2020': 105137,
 'blacklivesmatter': 112795,
 'mikepence': 483128,
 'debates2020': 195579,
 'kamalaharrisvp': 401348,
 'riots': 631892,
 'hunoterbidden': 349134,
 'aoc': 65231,
 'ilhanomar': 355775,
 'gavinnewsom': 298920,
 'tedwheeler': 719416,
 'gretchenwhitmer': 319454,
 'gtgt': 321651,
 'begining': 95714,
 'going': 312942,
 'worse': 822120,
 'marcinphoenix': 463826,
 'ciaspygirl': 155023,
 'trumpliesamericansdie': 754036,
 'immunologist': 358073,
 'florian': 281761,
 'kramme': 418756,
 'herdimmunity': 337238,
 'natural': 507992,
 'infections': 363927,
 'strategy': 700231,
 'indication': 362556,
 'governmenot': 316247,
 'malfunction': 460123,
 'conotrol': 170789,
 'break': 122914,
 'lost': 448946,
 'lives': 444015,
 'rudygiuliani': 639605,
 'presssec': 588238,
 'condemned': 167408,
 'cc': 143178,
 'american': 52703,
 'lets': 436092,
 'redo': 616428,
 'trumps': 756060,
 'bidencares': 104372,
 'bidenwillcrushcovid': 107476,
 'bidenharris2020tosaveamerica': 105180,
 'quoted': 604660,
 'xu': 826338,
 'russia': 640872,
 'putin': 600274,
 'north': 522597,
 'korea': 417599,
 'kim': 411349,
 'leaders': 431554,
 '100': 1129,
 'menotion': 477118,
 'allied': 48165,
 'counotries': 177091,
 'cnn': 160118,
 'andersoncooper': 57807,
 'trumpisunfitforoffice': 753418,
 'commenots': 164376,
 'understand': 770739,
 'ruthless': 641423,
 'hunoterbiden': 349136,
 'bidenharris': 105133,
 'trumppence2020': 755329,
 'wto': 823872,
 'coronavirus': 174589,
 'trade': 743075,
 'shame': 667557,
 'nikkihaley': 517452,
 'ridiculous': 630216,
 'ad': 34845,
 'lindseygrahamsc': 441827,
 'saying': 652038,
 'fighting': 276950,
 'lady': 424809,
 'retnot': 626441,
 'power': 582794,
 'spineless': 689647,
 'comes': 163848,
 'conotrolling': 170875,
 'lindesy': 441737,
 'sc': 652946,
 'southcarolina': 686804,
 'votejaimeharrisonsc': 798948,
 'karatblood': 403192,
 'kazeplaysjc': 406319,
 'grab': 316823,
 'balls': 88120,
 'chuck': 154260,
 'bastard': 91330,
 'door': 222841,
 'onoto': 541189,
 'pennsylvaniaavenue': 562275,
 'form': 285127,
 'line': 441880,
 'everybody': 261599,
 'gets': 305882,
 'kick': 410345,
 'nuts': 529716,
 'note': 524494,
 'person': 565259,
 'bidenharristosaveamerica': 105386,
 'votebluetosaveourdemocracy': 798266,
 'le': 431447,
 'pdt': 559878,
 'nous': 527366,
 'ne': 509686,
 'sommes': 683747,
 'pas': 555944,
 'venus': 785915,
 'daussi': 192149,
 'loin': 446479,
 'avons': 82062,
 'surmonoté': 707475,
 'tout': 742343,
 'cela': 144035,
 'et': 258968,
 'battus': 91954,
 'si': 671537,
 'duremenot': 230121,
 'pour': 582509,
 'rendre': 621130,
 'notre': 526187,
 'pays': 559303,
 'au': 77950,
 'marais': 463327,
 'dépravé': 232874,
 'washington': 806409,
 'ice': 352595,
 'cube': 184092,
 'teaming': 718572,
 'work': 821276,
 'developing': 209239,
 'platinum': 572903,
 'plan': 572234,
 '2020election': 8826,
 'africanamericans': 38881,
 'blackamerican': 112414,
 'diamondandsilk': 210976,
 'featurenotwjlive': 273624,
 'race': 605333,
 'twjreports': 764436,
 'usnews': 778710,
 'mariestrella': 464853,
 'carmenacoleman': 139667,
 'hugovenotura': 348119,
 'marvinicio': 467042,
 'micaelatacconi': 480476,
 'otro': 546685,
 'jonrón': 393532,
 'nos': 522813,
 'olvide': 538613,
 'ahí': 41844,
 'también': 714963,
 'están': 258870,
 'involucrados': 371448,
 'los': 448426,
 'clinotoncrimefamily': 158885,
 'mencionar': 476459,
 'algunos': 46444,
 'floridaman': 281894,
 'lemme': 434475,
 'make': 458928,
 'sure': 707244,
 'sooo': 684744,
 'checks': 148944,
 'notes': 524677,
 '1st': 7684,
 'supporter': 706528,
 'vandalized': 783075,
 'property': 593607,
 '2nd': 13979,
 'believed': 97197,
 'making': 459571,
 'look': 447346,
 'racist': 605783,
 'blacksfortrump': 113102,
 'blackvoicesfortrump': 113221,
 'bidenisaracist': 105690,
 'blackfathersmatter': 112613,
 'blackvotersmatter': 113234,
 'ticket': 734124,
 'somebody': 683504,
 'democratsaredestroyingamerica': 201863,
 'chrisdjackson': 153128,
 'new': 513369,
 'respect': 624943,
 'scaramucci': 653276,
 'vile': 792369,
 'hateful': 332260,
 'ugliness': 767166,
 'lawandorder': 430587,
 'owe': 549071,
 '420': 17695,
 'dollars': 221986,
 'terms': 722249,
 'beholden': 96135,
 'demand': 200440,
 'trumpisanationalsecurityrisk': 752724,
 'followthemoney': 283535,
 'trumptaxcheat': 757361,
 'trumptaxfraud': 757370,
 'biggest': 108305,
 'liar': 437720,
 'evner': 262366,
 'jimjordan': 387516,
 'devinnunes': 209458,
 'mattgaetz': 469545,
 'johncornyn': 391875,
 'billbarr': 109515,
 'rest': 625487,
 'railed': 607321,
 'indicted': 362601,
 'jail': 379809,
 'perpetuating': 564938,
 'fraud': 288619,
 'perfectly': 564035,
 'legal': 433266,
 'unmasking': 773205,
 'general': 302161,
 'flynn': 282575,
 'lockthemup': 445988,
 'emperor': 247047,
 'clothes': 159330,
 'end': 248259,
 'icecube': 352643,
 'thedemocrats': 726441,
 'care': 138818,
 'toe': 738356,
 'follow': 283390,
 'narrative': 506486,
 'help': 336309,
 'served': 664744,
 'claim': 157091,
 'policies': 576503,
 'helped': 336330,
 'credit': 180529,
 'davidjharrisjr': 192761,
 'demsarecorrupt': 203031,
 'fourmoreyears': 286538,
 'theweek': 731009,
 'penn': 562129,
 'saved': 651313,
 'suburbia': 703097,
 'rid': 630045,
 'regulation': 618596,
 'disaster': 215821,
 'really': 612731,
 'unfair': 771241,
 'gotten': 315905,
 'lot': 449131,
 'damn': 189080,
 'pennsylvania': 562252,
 'idenotify': 353291,
 'suppose': 706807,
 'feel': 274059,
 'badly': 86270,
 'claimed': 157104,
 'critics': 181884,
 'bow': 121000,
 'feet': 274207,
 'notald': 523246,
 'sorry': 685512,
 'snakes': 680033,
 'deserve': 206473,
 'joyannreid': 394974,
 'tearing': 718850,
 'hunoterbidenemails': 349170,
 'clinoton': 158862,
 'manwomanpersoncameratv': 462973,
 'rockpaperscissors': 635234,
 'handsfacespace': 329391,
 'votejoebiden': 798961,
 'uselection': 777766,
 'pence': 561764,
 'vs': 801483,
 'harris': 331151,
 'drugstest': 227586,
 'trumpdrugs': 750862,
 'bidendrugs': 104693,
 'injectdisinfectanot': 365704,
 'powerfullight': 582872,
 'skin': 676742,
 'track': 742965,
 'talking': 714557,
 '80': 24898,
 'researchers': 624279,
 'warn': 805990,
 'socalled': 681168,
 'approach': 67061,
 'managing': 460977,
 'dangerous': 189678,
 'fallacy': 269154,
 'unsupported': 774466,
 'scienotific': 656426,
 'evidence': 262006,
 'wcph2020': 807828,
 'wearamask': 808198,
 'maskup': 467983,
 'negarmortazavi': 510510,
 'according': 32605,
 'niece': 516395,
 'metally': 478925,
 'ask': 73857,
 'menotal': 476937,
 'case': 140847,
 'behave': 95962,
 'feeling': 274080,
 'ist': 375111,
 'gt': 321338,
 'pest': 566113,
 'united': 772435,
 'sht': 671154,
 'simple': 673968,
 'question': 603812,
 'uspoli': 778839,
 'mapoli': 463185,
 '216000': 10074,
 'deaths': 195263,
 'magically': 457578,
 'amazing': 51438,
 'need': 510168,
 'fear': 273481,
 'covidiots': 178722,
 'sad': 644383,
 'fools': 283984,
 'believe': 97188,
 'trumpfakedcovid': 751331,
 'trumpfailed': 751297,
 'dumptrump': 229431,
 'trumpcrimefamily': 750407,
 'education': 237275,
 'thought': 733028,
 'brick': 124159,
 'wall': 804956,
 'notaldjtrumpjr': 523403,
 'notaldtrumpjr': 523746,
 'debate2020': 195428,
 'foxnews': 286925,
 'trump2020landslide': 748857,
 'suburban': 703052,
 'women': 820146,
 'pleaded': 573300,
 'neighborhood': 510872,
 'openly': 542138,
 'votes': 799774,
 'berniesanders': 99678,
 'biden2020': 103802,
 'harris2020': 331153,
 'registertovote': 618291,
 'glennkirschner2': 310318,
 'basically': 91061,
 'experimenot': 264306,
 'failed': 267893,
 'mercy': 477812,
 'theses': 730230,
 'agenda': 39628,
 'let': 435903,
 'pretend': 588475,
 'wish': 818425,
 'list': 443160,
 'hiphop': 341120,
 'legend': 433499,
 'team': 718446,
 'thoughts': 733088,
 'trending': 745948,
 'social': 681273,
 'media': 472995,
 'credible': 180520,
 'news': 513895,
 'outlets': 547574,
 'showing': 670800,
 'crisis': 181559,
 'mongering': 492115,
 'bit': 111370,
 'epitome': 252960,
 'fake': 268413,
 'cnnfakenews': 160219,
 'dcdrnoto': 194167,
 'using': 778377,
 'doj': 221568,
 'retaliate': 626204,
 'political': 576850,
 'opponenots': 542979,
 'wh': 812350,
 'staffers': 692870,
 'fuck': 292431,
 'votebidenharris2020': 798030,
 'theinfected': 727427,
 'potus': 582196,
 'doofus': 222718,
 'fall': 269142,
 'lie': 439149,
 'remember': 620398,
 'covid': 178140,
 'trumptrnot': 757768,
 'block': 114747,
 'screenshotting': 657771,
 'hunoter': 349110,
 'story': 699564,
 'retweet': 626935,
 'screenshots': 657768,
 'texas': 723572,
 'patriot': 557612,
 'followers': 283430,
 'northcarolina': 522623,
 'ncpol': 509390,
 'cbp': 143040,
 'usbp': 777373,
 'stoppingtheillegalflowofdrugs': 699038,
 'savingamericanlives': 651649,
 'greatworkcbp': 318654,
 'thankyou': 724562,
 'sounds': 686514,
 'newsmax': 514250,
 'sleepyjoe': 677937,
 'bad': 85937,
 'decision': 196733,
 'americans': 53304,
 'value': 782805,
 'truth': 759101,
 'stop': 698588,
 'propaganda': 593360,
 'corruption': 175501,
 'votebiden': 798008,
 'bankruptbillionaire': 89087,
 'taxcheattrump': 717350,
 'cheaterinchief': 148688,
 'newjersey': 513652,
 'says': 652223,
 'trumpisanationaldisgrace': 752720,
 'cousin': 177909,
 'posted': 581326,
 'message': 478628,
 'doctors': 220721,
 'office': 534877,
 'bs': 126792,
 'nerve': 512023,
 'thehill': 727301,
 'common': 164638,
 'knowledge': 415153,
 'concerns': 166974,
 'babygate': 85233,
 'bunkerboy': 129187,
 'nbitch': 509174,
 'captnotconotagious': 138239,
 'realizing': 612557,
 'plot': 573889,
 'destroy': 208106,
 'usps': 779032,
 'stupid': 701844,
 'idea': 353078,
 'collusion': 162669,
 'stripping': 700941,
 'aca': 31815,
 'cv': 186012,
 'spread': 690889,
 '220k': 10335,
 'dead': 194549,
 'unemployed': 771056,
 'small': 678893,
 'businesses': 130262,
 'struggling': 701261,
 'nk': 518644,
 'escalating': 255940,
 '430': 17848,
 'owed': 549072,
 'taxesfinancials': 717419,
 'obstructed': 532962,
 'voter': 799341,
 'suppression': 706866,
 'af': 37649,
 'abt': 31392,
 'mind': 484821,
 'script': 657861,
 'stuff': 701653,
 'isyou': 375720,
 'wanot': 805538,
 'wheres': 813375,
 'things': 731622,
 'goneawry': 313716,
 'broadcasted': 125289,
 'thank': 724417,
 'mr': 496160,
 'freedom': 289519,
 'kag2020': 400190,
 'zillalady': 838358,
 'maddow': 456214,
 'lawrence': 430750,
 'chrislhayes': 153249,
 '11thhour': 2990,
 'govkristinoem': 316444,
 'kristinoem': 419464,
 'fox': 286721,
 'state': 694245,
 'differenot': 212772,
 'promoting': 593038,
 'wenot': 810969,
 'called': 134846,
 'midnight': 482064,
 'master': 468543,
 'hello': 336110,
 'image': 356882,
 'background': 85623,
 'removal': 620733,
 'retuch': 626834,
 'resize': 624780,
 'gig': 307933,
 'fiverr': 280038,
 'photoshop': 568552,
 'backgroundremoval': 85637,
 'removebackground': 620747,
 'retouching': 626531,
 'lowa': 450231,
 'desmoines': 207126,
 'cutout': 185868,
 'crocs': 182034,
 'dexter': 209883,
 'showtime': 670907,
 'michaelchall': 480623,
 'twittersafety': 764220,
 'reciaimthenet': 614612,
 'inoterference': 367350,
 'ongoing': 540514,
 'democratic': 201569,
 'proofs': 593325,
 'hacked': 326744,
 'materials': 468932,
 'giuliani': 309372,
 'man': 460926,
 'respectable': 624950,
 'reputationis': 623983,
 'prison': 589887,
 'involved': 371468,
 'shady': 667232,
 'enoterprisesukrnote': 250787,
 'republicans': 623389,
 'forged': 284944,
 'charliekirk11': 147991,
 'truthmatters': 759274,
 'proved': 595217,
 'dangerously': 189709,
 'incompetenot': 361198,
 'faced': 266952,
 'world': 821593,
 'safer': 644786,
 'vp': 801076,
 'makes': 459399,
 'great': 318381,
 'partnerships': 555623,
 'wanots': 805604,
 'powerful': 582867,
 'counotry': 177128,
 'possible': 581060,
 'global': 310496,
 'votingbiden': 800477,
 'south': 686757,
 'carolina': 139866,
 '10142020': 1524,
 'polling': 578055,
 'average': 81637,
 '510': 20218,
 '01': 164,
 'yesterday': 830795,
 '438': 17928,
 '02': 257,
 'probabiity': 590401,
 '887': 25789,
 '03': 336,
 '113': 2495,
 'source': 686591,
 'fivethirtyeight': 280074,
 'electionotwitter': 242090,
```