Covid-19 Vaccine Sentiment Analysis

Goal: To examine tweets recorded about the Covid-19 vaccine to analyze the sentiments of people for the vaccine.

In [12]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [106]:
```
df = pd.read_csv("vaccination_tweets.csv")
```

In [4]:
```
df.head()
```

Out[4]:

	id	user_name	user_location	user_description	user_created	user_followers	user_friends	user_favourites
0	1340539111971516416	Rachel Roh	La Crescenta-Montrose, CA	Aggregator of Asian American news; scanning di...	2009-04-08 17:52:46	405	1692	3247
1	1338158543359250433	Albert Fong	San Francisco, CA	Marketing dude, tech geek, heavy metal & '80s ...	2009-09-21 15:27:30	834	666	178
2	1337858199140118533	eli🇱🇹🇪🇺👌	Your Bed	heil, hydra 🖐☺	2020-06-25 23:30:28	10	88	155
3	1337855739918835717	Charles Adler	Vancouver, BC - Canada	Hosting "CharlesAdlerTonight" Global News Radi...	2008-09-10 11:28:53	49165	3933	21853
4	1337854064604966912	Citizen News Channel	NaN	Citizen News Channel bringing you an alternati...	2020-04-23 17:58:42	152	580	1473

In [107]:
```
df.columns
```

Out[107]:
```
Index(['id', 'user_name', 'user_location', 'user_description', 'user_created',
       'user_followers', 'user_friends', 'user_favourites', 'user_verified',
       'date', 'text', 'hashtags', 'source', 'retweets', 'favorites',
       'is_retweet'],
      dtype='object')
```

In [7]:
```
df.dtypes
```

Out[7]:
```
id                   int64
user_name           object
user_location       object
user_description    object
user_created        object
user_followers       int64
user_friends         int64
user_favourites      int64
user_verified         bool
date                object
text                object
hashtags            object
source              object
retweets             int64
favorites            int64
is_retweet            bool
dtype: object
```

In [108]:
```
df['source'].unique()
```

Out[108]:
```
array(['Twitter for Android', 'Twitter Web App', 'Twitter for iPhone',
       'TweetDeck', 'Buffer', 'Twitter for iPad', 'Twitter Media Studio',
       'ThreadReaderApp', 'Instagram', 'SocialFlow', 'Hootsuite Inc.',
       'LinkedIn', 'Twitter for Mac', '24liveblog', 'Publer ', 'IFTTT',
       'Socialbakers', 'Falcon Social Media Management', 'Echobox',
       'Microsoft Power Platform', 'Nonli', 'Sendible', nan, 'EastMojo',
       'Twitter Media Studio - LiveCut', 'Tweetbot for Mac', 'GT_Backend',
       'SocialNewsDesk', 'Sprout Social', 'TweetCaster for Android',
       'UberSocial for Android', 'Blog2Social APP', 'WordPress.com',
       'Paper.li', 'News Users', 'ETRetail.com', 'Tweetbot for iOS',
       'Sprinklr Publishing', 'CoSchedule', 'Hocalwire Social Share',
       'Canva', 'Nelio Content', 'Flying Eze', 'News Medical', 'dlvr.it',
       'Fenix 2', 'Imminent News', 'Sprinklr',
       'Salesforce - Social Studio', 'Echofon', 'Article Tweetbot',
       'Tumblr', 'Loomly', 'The Tweeted Times',
       'SEMrush Social Media Tool', 'Hash4trends', 'Threader_client',
       'Revive Social App', 'Business Upturn', 'Hypefury', 'LaterMedia',
       'Zoho Social', 'BizToc News', 'BizToc', 'Social-Pull-Staging'],
      dtype=object)
```

In [9]:
```
df['source'].value_counts()
```

Out[9]:
```
Twitter for Android    58722
Twitter Web App        57119
Twitter for iPhone     49479
cowin_vaccine_app      11670
CowinAlertsBot         11276
                       ...  
Bitly                      1
Injunnie                   1
todayinnevada              1
Gravity Forever            1
Vitamin.Rush               1
Name: source, Length: 379, dtype: int64
```

In [109]:
```
df['source'].value_counts().nlargest(30)
```

Out[109]:
```
Twitter for iPhone                3579
Twitter Web App                   3258
Twitter for Android               2716
TweetDeck                          618
Twitter for iPad                   272
Instagram                          180
Buffer                              69
Hootsuite Inc.                      66
IFTTT                               22
Blog2Social APP                     18
LinkedIn                            17
Microsoft Power Platform            15
Tweetbot for iOS                    12
Nonli                               10
Revive Social App                   10
dlvr.it                              8
Paper.li                             8
WordPress.com                        8
Twitter for Mac                      8
24liveblog                           7
SocialFlow                           7
Sprinklr Publishing                  7
ETRetail.com                         7
Sprout Social                        7
Sendible                             7
Article Tweetbot                     6
Salesforce - Social Studio           5
Twitter Media Studio                 5
Flying Eze                           5
Twitter Media Studio - LiveCut       4
Name: source, dtype: int64
```

In [110]:
```
plt.figure(figsize=(20,10))
df['source'].value_counts().nlargest(30).plot(kind='bar')
plt.xticks(rotation=45)
plt.show()
```

Text Analysis of tweets

In [111]:
```
import neattext.functions as nfx
```

In [112]:
```
df['text'].iloc[2]
```

Out[112]:
```
'#coronavirus #SputnikV #AstraZeneca #PfizerBioNTech #Moderna #Covid_19 Russian vaccine is created to last 2-4 years… https://t.co/ieYlCKBr8P'
```

In [113]:
```
df['text'].apply(nfx.extract_hashtags)
```

Out[113]:
```
0                                    [#PfizerBioNTech]
1                                                   []
2         [#coronavirus, #SputnikV, #AstraZeneca, #Pfize...
3                                                   []
4              [#whereareallthesickpeople, #PfizerBioNTech…]
                              ...
11015                                               []
11016                                [#PfizerBioNTech]
11017                                               []
11018             [#PfizerBioNTech, #Paxlovid, #COVID19]
11019                                               []
Name: text, Length: 11020, dtype: object
```

In [114]:
```
df['extracted_hashtags'] = df['text'].apply(nfx.extract_hashtags)
```

In [115]:
```
df
```

Out[115]:

	id	user_name	user_location	user_description	user_created	user_followers	user_friends
0	1340539111971516416	Rachel Roh	La Crescenta-Montrose, CA	Aggregator of Asian American news; scanning di...	2009-04-08 17:52:46	405	1692
1	1338158543359250433	Albert Fong	San Francisco, CA	Marketing dude, tech geek, heavy metal & '80s ...	2009-09-21 15:27:30	834	666
2	1337858199140118533	eli🇱🇹🇪🇺👌	Your Bed	heil, hydra 🖐☺	2020-06-25 23:30:28	10	88
3	1337855739918835717	Charles Adler	Vancouver, BC - Canada	Hosting "CharlesAdlerTonight" Global News Radi...	2008-09-10 11:28:53	49165	3933
4	1337854064604966912	Citizen News Channel	NaN	Citizen News Channel bringing you an alternati...	2020-04-23 17:58:42	152	580
...	...	...	...	...	...	...	...
11015	1460932683630977025	theSun	Petaling Jaya	Breaking news updates from the website of Mala...	2009-07-29 01:58:44	151937	465
11016	1460809350792896516	Sujan	NaN	•Napricorn• •It's jus...	2013-01-28 11:48:20	4259	622
11017	1460581918295031809	theSun	Petaling Jaya	Breaking news updates from the website of Mala...	2009-07-29 01:58:44	151937	465
11018	1460344277951782922	Aleksejs Jackovs	Ireland	#Scientificresearcher in #InformationTechnolog...	2012-08-13 13:47:03	1	38
11019	1460208097956225028	theSun	Petaling Jaya	Breaking news updates from the website of Mala...	2009-07-29 01:58:44	151937	465

11020 rows × 17 columns

In [116]:
```
#remove hasgtags
df['clean_tweet'] = df['text'].apply(nfx.remove_hashtags)
```

In [117]:
```
df[['text','clean_tweet']]
```

Out[117]:

	text	clean_tweet
0	Same folks said daikon paste could treat a cyt...	Same folks said daikon paste could treat a cyt...
1	While the world has been on the wrong side of ...	While the world has been on the wrong side of ...
2	#coronavirus #SputnikV #AstraZeneca #PfizerBio...	Russian vaccine is created to last...
3	Facts are immutable, Senator, even when you're...	Facts are immutable, Senator, even when you're...
4	Explain to me again why we need a vaccine @Bor...	Explain to me again why we need a vaccine @Bor...
...	...	...
11015	The number of Covid-19 cases today, 17 Novembe...	The number of Covid-19 cases today, 17 Novembe...
11016	First dose ✅\n#PfizerBioNTech	First dose ✅\n
11017	The number of Covid-19 cases today, 16 Novembe...	The number of Covid-19 cases today, 16 Novembe...
11018	#PfizerBioNTech has developed new pill #Paxlov...	has developed new pill that increased the ...
11019	The number of Covid-19 cases today, 15 Novembe...	The number of Covid-19 cases today, 15 Novembe...

11020 rows × 2 columns

In [118]:
```
df['clean_tweet'] = df['clean_tweet'].apply(lambda x:nfx.remove_userhandles(x))
```

In [119]:
```
#remove multiple white spaces
df['clean_tweet'] = df['clean_tweet'].apply(nfx.remove_multiple_spaces)
```

In [120]:
```
df['clean_tweet']
```

Out[120]:
```
0        Same folks said daikon paste could treat a cyt...
1        While the world has been on the wrong side of ...
2         Russian vaccine is created to last 2-4 years…...
3        Facts are immutable, Senator, even when you're...
4        Explain to me again why we need a vaccine http...
                              ...
11015    The number of Covid-19 cases today, 17 Novembe...
11016                                         First dose ✅
11017    The number of Covid-19 cases today, 16 Novembe...
11018     has developed new pill that increased the cha...
11019    The number of Covid-19 cases today, 15 Novembe...
Name: clean_tweet, Length: 11020, dtype: object
```

In [121]:
```
#remove urls
df['clean_tweet'] = df['clean_tweet'].apply(nfx.remove_urls)
```

In [122]:
```
#remove punctuations
df['clean_tweet'] = df['clean_tweet'].apply(nfx.remove_puncts)
```

In [123]:
```
df[['text','clean_tweet']]
```

Out[123]:

	text	clean_tweet
0	Same folks said daikon paste could treat a cyt...	Same folks said daikon paste could treat a cyt...
1	While the world has been on the wrong side of ...	While the world has been on the wrong side of ...
2	#coronavirus #SputnikV #AstraZeneca #PfizerBio...	Russian vaccine is created to last 24 years…
3	Facts are immutable, Senator, even when you're...	Facts are immutable Senator even when youre no...
4	Explain to me again why we need a vaccine @Bor...	Explain to me again why we need a vaccine
...	...	...
11015	The number of Covid-19 cases today, 17 Novembe...	The number of Covid19 cases today 17 November ...
11016	First dose ✅\n#PfizerBioNTech	First dose ✅
11017	The number of Covid-19 cases today, 16 Novembe...	The number of Covid19 cases today 16 November ...
11018	#PfizerBioNTech has developed new pill #Paxlov...	has developed new pill that increased the cha...
11019	The number of Covid-19 cases today, 15 Novembe...	The number of Covid19 cases today 15 November ...

11020 rows × 2 columns

Sentiment Analysis

In [124]:
```
from textblob import TextBlob
```

In [159]:
```
def get_sentiment(text):
    blob = TextBlob(text)
    sentiment_polarity = blob.sentiment.polarity
    sentiment_subjectivity = blob.sentiment.subjectivity
    if sentiment_polarity >0:
        sentiment_label = 'Positive'
    elif sentiment_polarity <0:
        sentiment_label = 'Negative'
    else:
        sentiment_label = 'Neutral'
    result = {'polarity':sentiment_polarity,
             'subjectivity':sentiment_subjectivity,
             'sentiment':sentiment_label}
    return result
```

In [160]:
```
exl = df['clean_tweet'].iloc[0]
```

In [161]:
```
get_sentiment(exl)
```

Out[161]:
```
{'polarity': 0.0, 'subjectivity': 0.125, 'sentiment': 'Neutral'}
```

In [162]:
```
df['sentiment_results'] = df['clean_tweet'].apply(get_sentiment)
```

In [210]:
```
pd.json_normalize(df['sentiment_results'])
```

Out[210]:

	polarity	subjectivity	sentiment
0	0.000000	0.125000	Neutral
1	-0.500000	0.900000	Negative
2	0.000000	0.033333	Neutral
3	-0.050000	0.550000	Negative
4	0.000000	0.000000	Neutral
...	...	...	...
11015	0.000000	0.000000	Neutral
11016	0.250000	0.333333	Positive
11017	0.000000	0.000000	Neutral
11018	-0.121212	0.501515	Negative
11019	0.000000	0.000000	Neutral

11020 rows × 3 columns

In [212]:
```
df
```

Out[212]:

	id	user_name	user_location	user_description	user_created	user_followers	user_friends
0	1340539111971516416	Rachel Roh	La Crescenta-Montrose, CA	Aggregator of Asian American news; scanning di...	2009-04-08 17:52:46	405	1692
1	1338158543359250433	Albert Fong	San Francisco, CA	Marketing dude, tech geek, heavy metal & '80s ...	2009-09-21 15:27:30	834	666
2	1337858199140118533	eli🇱🇹🇪🇺👌	Your Bed	heil, hydra 🖐☺	2020-06-25 23:30:28	10	88
3	1337855739918835717	Charles Adler	Vancouver, BC - Canada	Hosting "CharlesAdlerTonight" Global News Radi...	2008-09-10 11:28:53	49165	3933
4	1337854064604966912	Citizen News Channel	NaN	Citizen News Channel bringing you an alternati...	2020-04-23 17:58:42	152	580
...	...	...	...	...	...	...	...
11015	1460932683630977025	theSun	Petaling Jaya	Breaking news updates from the website of Mala...	2009-07-29 01:58:44	151937	465
11016	1460809350792896516	Sujan	NaN	•Napricorn• •It's jus...	2013-01-28 11:48:20	4259	622
11017	1460581918295031809	theSun	Petaling Jaya	Breaking news updates from the website of Mala...	2009-07-29 01:58:44	151937	465
11018	1460344277951782922	Aleksejs Jackovs	Ireland	#Scientificresearcher in #InformationTechnolog...	2012-08-13 13:47:03	1	38
11019	1460208097956225028	theSun	Petaling Jaya	Breaking news updates from the website of Mala...	2009-07-29 01:58:44	151937	465

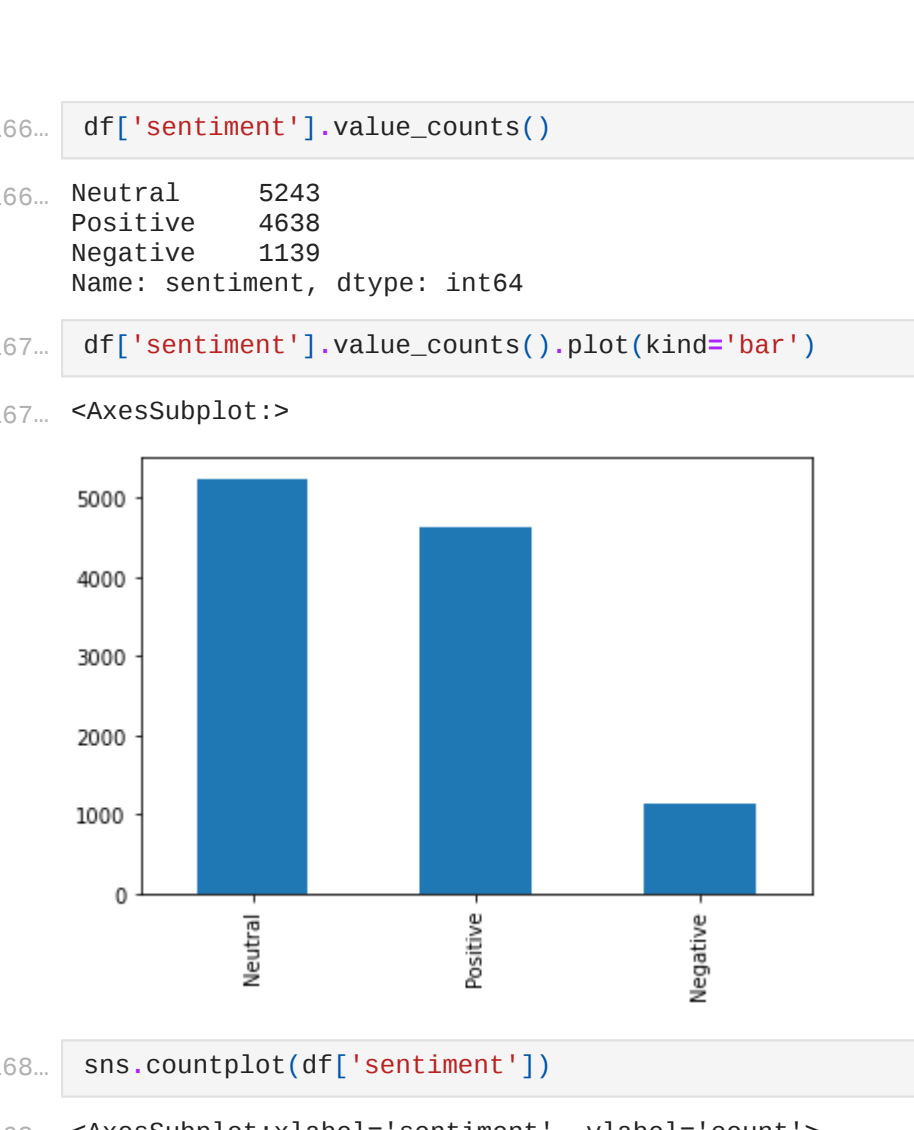
11020 rows × 22 columns

In [166]:
```
df['sentiment'].value_counts()
```

Out[166]:
```
Neutral     5243
Positive    4638
Negative    1139
Name: sentiment, dtype: int64
```

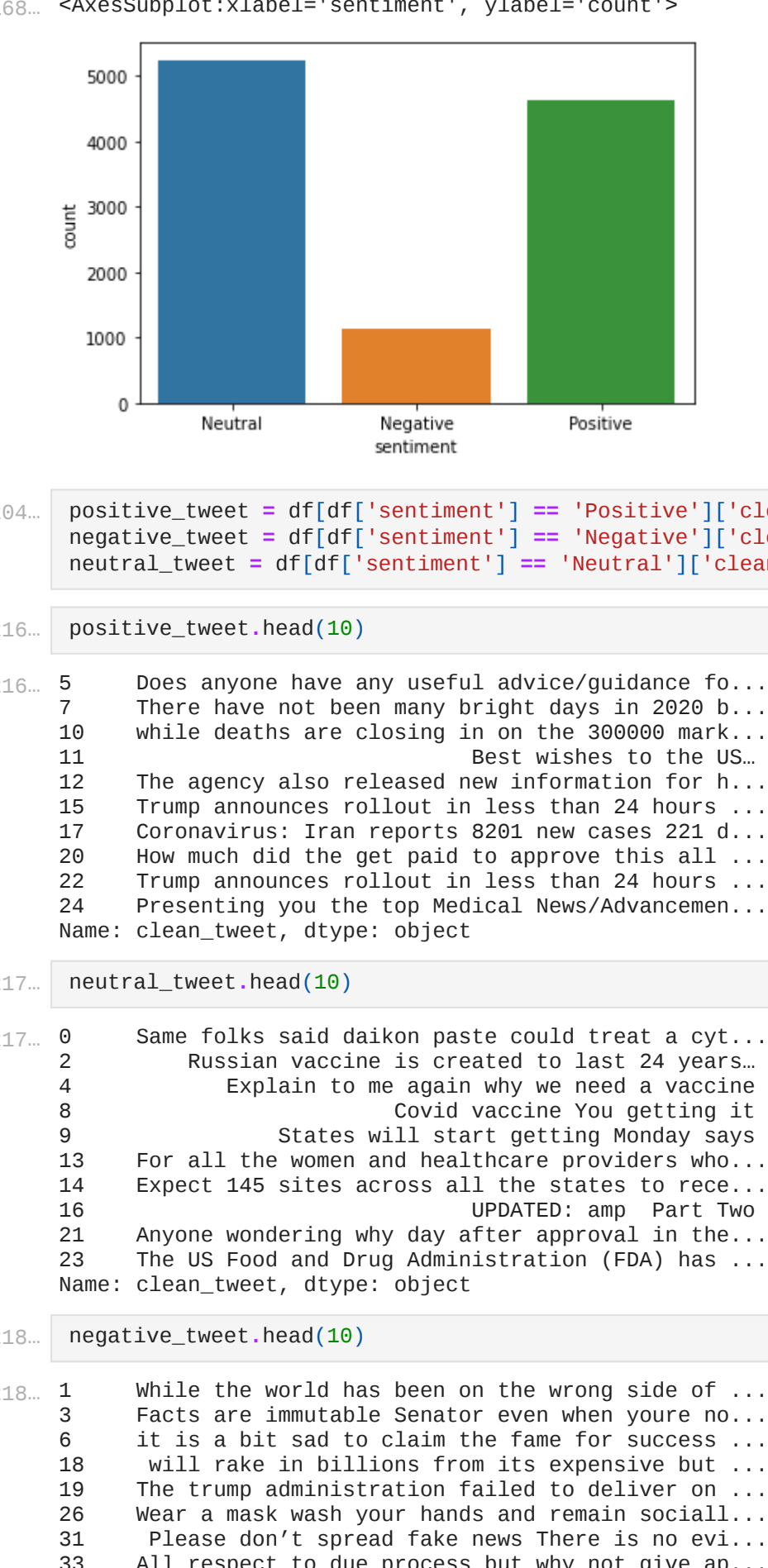
In [167]:
```
df['sentiment'].value_counts().plot(kind='bar')
```

Out[167]: `<AxesSubplot:>`



In [168]:
```
sns.countplot(df['sentiment'])
```

Out[168]: `<AxesSubplot:xlabel='sentiment', ylabel='count'>`



In [204]:
```
positive_tweet = df[df['sentiment'] == 'Positive']['clean_tweet']
negative_tweet = df[df['sentiment'] == 'Negative']['clean_tweet']
neutral_tweet = df[df['sentiment'] == 'Neutral']['clean_tweet']
```

In [216]:
```
positive_tweet.head(10)
```

Out[216]:
```
5     Does anyone have any useful advice/guidance fo...
7     There have not been many bright days in 2020 b...
10    while deaths are closing in on the 300000 mark...
11                               Best wishes to the US…
12    The agency also released new information for h...
15    Trump announces rollout in less than 24 hours ...
17    Coronavirus: Iran reports 8201 new cases 221 d...
20    How much did the get paid to approve this all ...
22    Trump announces rollout in less than 24 hours ...
24    Presenting you the top Medical News/Advancemen...
Name: clean_tweet, dtype: object
```

In [217]:
```
neutral_tweet.head(10)
```

Out[217]:
```
0     Same folks said daikon paste could treat a cyt...
2         Russian vaccine is created to last 24 years…
4            Explain to me again why we need a vaccine
8                         Covid vaccine You getting it
9                States will start getting Monday says
13    For all the women and healthcare providers who...
14    Expect 145 sites across all the states to rece...
16                               UPDATED: amp  Part Two
21    Anyone wondering why day after approval in the...
23    The US Food and Drug Administration (FDA) has ...
Name: clean_tweet, dtype: object
```

In [218]:
```
negative_tweet.head(10)
```

Out[218]:
```
1     While the world has been on the wrong side of ...
3     Facts are immutable Senator even when youre no...
6     it is a bit sad to claim the fame for success ...
18     will rake in billions from its expensive but ...
19    The trump administration failed to deliver on ...
26    Wear a mask wash your hands and remain sociall...
31     Please don’t spread fake news There is no evi...
33    All respect to due process but why not give ap...
39     check out table 23 of this and other tables t...
50    Tragedy Another life taken by corporate greed ...
Name: clean_tweet, dtype: object
```

In [224]:
```
positive_tweet_list = positive_tweet.apply(nfx.remove_stopwords).tolist()
negative_tweet_list = negative_tweet.apply(nfx.remove_stopwords).tolist()
neutral_tweet_list = neutral_tweet.apply(nfx.remove_stopwords).tolist()
```

In [175]:
```
pos_tokens = [token for line in positive_tweet_list for token in line.split()]
neg_tokens = [token for line in negative_tweet_list for token in line.split()]
neut_tokens = [token for line in neutral_tweet_list for token in line.split()]
```

In [177]:
```
#Get most commonest keywords
from collections import Counter
```

In [178]:
```
def get_tokens(docx,num=30):
    word_tokens = Counter(docx)
    most_common = word_tokens.most_common(num)
    result = dict(most_common)
    return result
```

In [179]:
```
get_tokens(pos_tokens)
```

Out[179]:
```
{'vaccine': 1355,
 'dose': 748,
 'today': 349,
 'Pfizer': 278,
 'got': 273,
 'COVID19': 233,
 'effective': 215,
 'amp': 210,
 'received': 209,
 'vaccines': 203,
 'doses': 184,
 'shot': 182,
 'new': 177,
 'second': 177,
 'people': 175,
 'vaccinated': 174,
 'vaccination': 163,
 'Vaccine': 163,
 'jab': 149,
 'Thanks': 147,
 '2nd': 147,
 'Got': 138,
 'good': 137,
 'Covid19': 136,
 '💉': 132,
 'Thank': 132,
 'day': 118,
 'the…': 109,
 '1st': 107,
 '2': 105}
```

In [180]:
```
most_common_pos_words = get_tokens(pos_tokens)
most_common_neg_words = get_tokens(neg_tokens)
most_common_neut_words = get_tokens(neut_tokens)
```

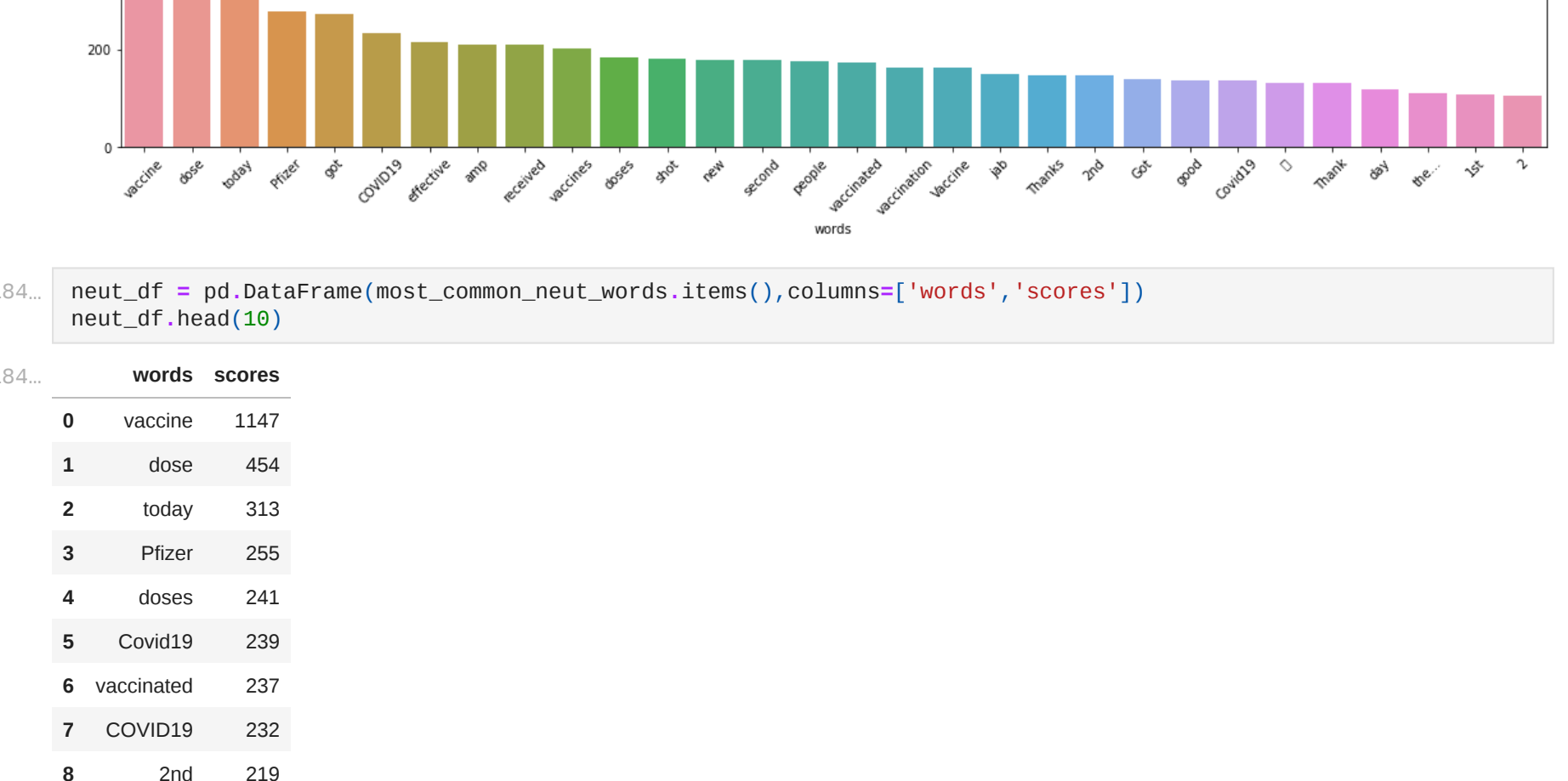
In [181]:
```
pos_df = pd.DataFrame(most_common_pos_words.items(),columns=['words','scores'])
```

In [182]:
```
pos_df.head(10)
```

Out[182]:

	words	scores
0	vaccine	1355
1	dose	748
2	today	349
3	Pfizer	278
4	got	273
5	COVID19	233
6	effective	215
7	amp	210
8	received	209
9	vaccines	203

In [183]:
```
plt.figure(figsize=(20,10))
sns.barplot(x='words',y='scores',data=pos_df)
plt.xticks(rotation=45)
plt.show()
```

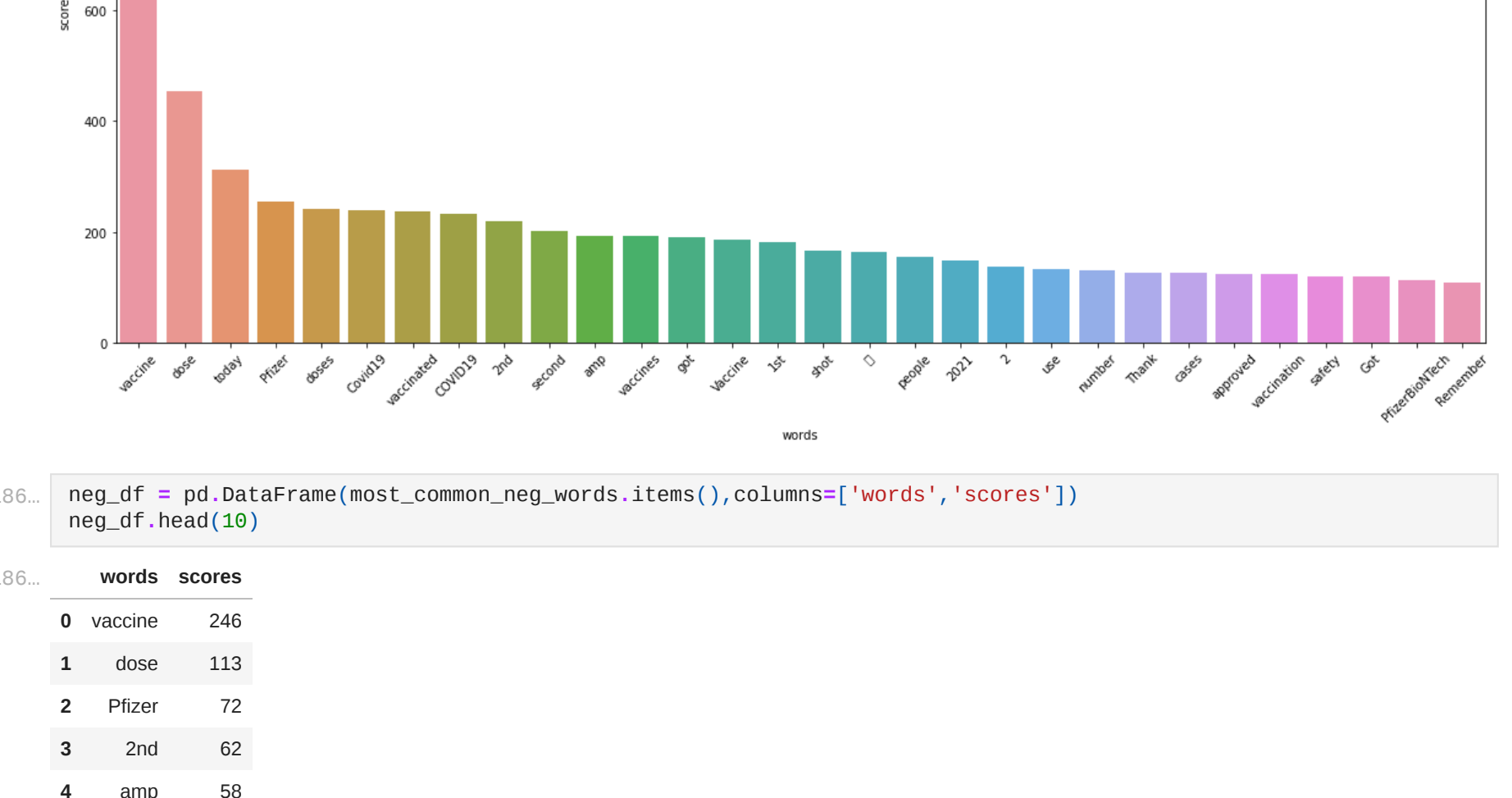


In [184]:
```
neut_df = pd.DataFrame(most_common_neut_words.items(),columns=['words','scores'])
neut_df.head(10)
```

Out[184]:

	words	scores
0	vaccine	1147
1	dose	454
2	today	313
3	Pfizer	255
4	doses	241
5	Covid19	239
6	vaccinated	237
7	COVID19	232
8	2nd	219
9	second	202

In [185]:
```
plt.figure(figsize=(20,10))
sns.barplot(x='words',y='scores',data=neut_df)
plt.xticks(rotation=45)
plt.show()
```



In [186]:
```
neg_df = pd.DataFrame(most_common_neg_words.items(),columns=['words','scores'])
neg_df.head(10)
```

Out[186]:

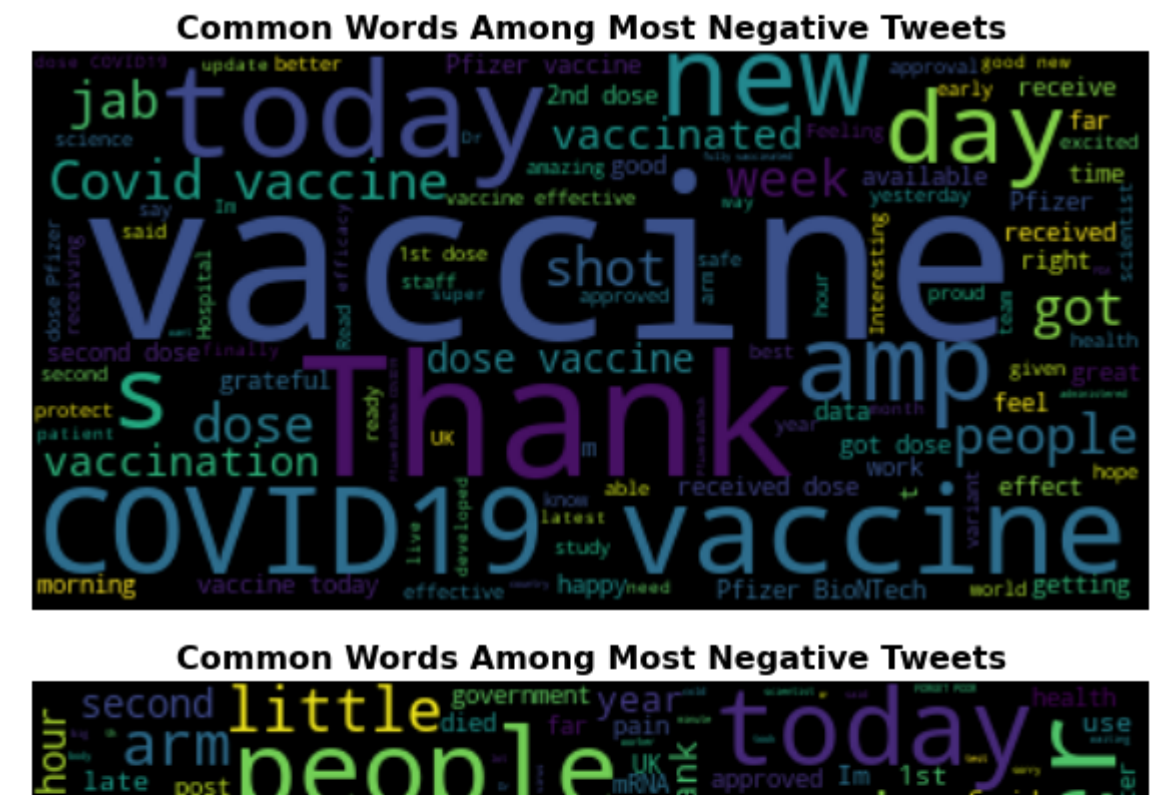
	words	scores
0	vaccine	246
1	dose	113
2	Pfizer	72
3	2nd	62
4	amp	58
5	arm	51
6	people	51
7	today	46
8	second	45
9	shot	43

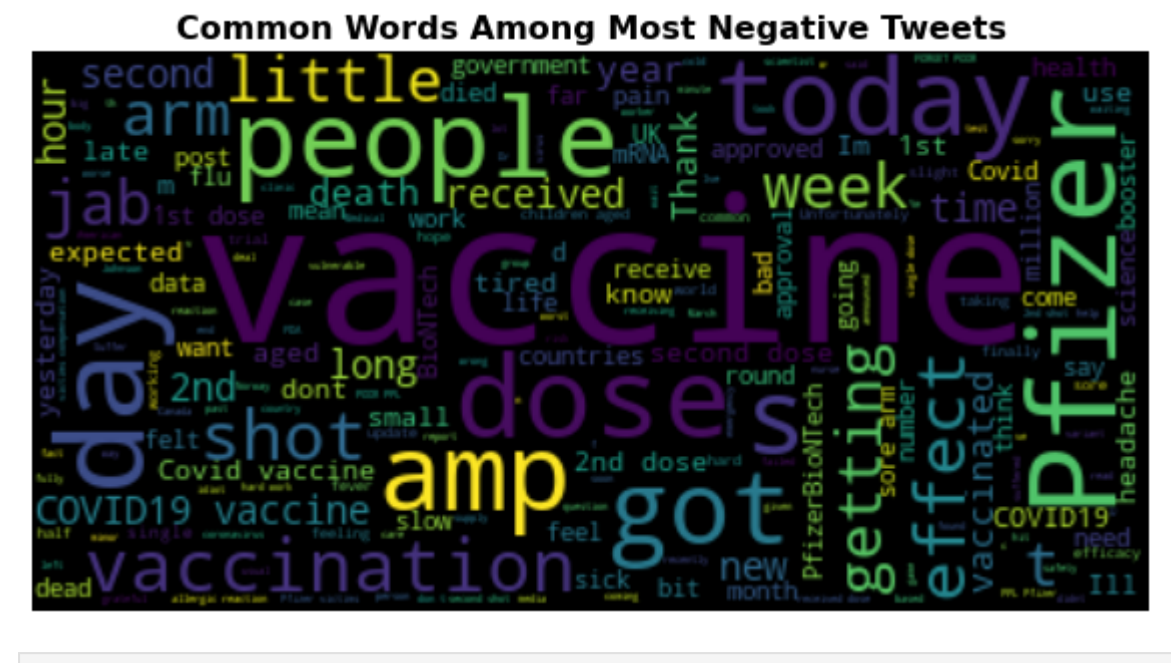
In [152]:
```
# Word Cloud
from wordcloud import WordCloud
```

In [213]:
```
def plot_wordcloud(docx):
    plt.figure(figsize=(10,10))
    plt.title('Common Words Among Most Positive Tweets',fontsize=16,fontweight='bold')
    plt.title('Common Words Among Most Negative Tweets',fontsize=16,fontweight='bold')
    mywordcloud = WordCloud().generate(docx)
    plt.imshow(mywordcloud,interpolation='bilinear')
    plt.axis('off')
    plt.show()
```

In [214]:
```
pos_docx = ' '.join(pos_tokens)
neg_docx = ' '.join(neg_tokens)
neut_docx = ' '.join(neut_tokens)
```

In [215]:
```
plot_wordcloud(pos_docx)
plot_wordcloud(neg_docx)
```





In [ ]: