In [ ]:

BEMM458J Final assignment

Student Number: 700013862

You are a business analyst at the marketing department of Coca Cola. There is an increasing debate on social media in relation to the negative impact of plastic consumption on the environment.

As of recently NGOs have started campaigning against Coca Cola and other multinationals.

General management needs you to conduct an analysis of recent conversations posted on Twitter for the purposes of determining the communication strategies followed by NGOs and how Coca Cola must engage on social media.

In [774…

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import statsmodels.formula.api as sm
from statsmodels.stats import diagnostic as diag
import re
from wordcloud import WordCloud


import statsmodels.api as sm
```

In [2]:

```
from IPython.core.display import display, HTML
display(HTML("<style>.container { width:100% !important; }</style>"))
```

In [3]:
```
pd.options.display.max_colwidth = 400
```

In [730…
```
ConversationsLean=pd.read_csv('../data/ConversationsLean.csv')
```

In [5]:
```
ConversationsLean.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 447 entries, 0 to 446
Data columns (total 31 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   Unnamed: 0                   447 non-null    int64  
 1   tweet.created_at             447 non-null    object 
 2   tweet.id                     447 non-null    float64
 3   tweet.full_text              447 non-null    object 
 4   tweet.entities               447 non-null    object 
 5   tweet.user.id                447 non-null    int64  
 6   tweet.user.screen_name       447 non-null    object 
 7   tweet.user.followers_count   447 non-null    int64  
 8   tweet.user.friends_count     447 non-null    int64  
 9   tweet.user.favourites_count  447 non-null    int64  
 10  tweet.user.statuses_count    447 non-null    int64  
 11  tweet.retweet_count          447 non-null    float64
 12  tweet.favorite_count         447 non-null    float64
 13  tweet.favorited              447 non-null    bool   
 14  tweet.retweeted              447 non-null    bool   
 15  tweet.lang                   447 non-null    object 
 16  fetchedAt                    447 non-null    object 
 17  tweet.full_text_clean        447 non-null    object 
 18  anger                        447 non-null    float64
 19  fear                         447 non-null    float64
 20  joy                          447 non-null    float64
 21  love                         447 non-null    float64
 22  sadness                      447 non-null    float64
 23  trust                        447 non-null    float64
 24  identity_hate                447 non-null    float64
 25  insult                       447 non-null    float64
 26  obscene                      447 non-null    float64
 27  severe_toxic                 447 non-null    float64
 28  threat                       447 non-null    float64
 29  toxic                        447 non-null    float64
 30  stakeholder                  440 non-null    object 
dtypes: bool(2), float64(15), int64(6), object(8)
memory usage: 102.3+ KB
```

In [6]:
```
ConversationsLean.iloc[50]
```

Out[6]:

```
Unnamed: 0
4
tweet.created_at
2019-12-17 19:50:37.000000
tweet.id
1.20703e+18
tweet.full_text
Stop plastic pollution at its source. #NoPttGlobalCracker  @GCNewsL @GovMikeDeWine protect our kids’ health and #SavetheOhioRiver
#PlanetOrPlastic #BreakFreeFromPlastic https://t.co/xYq2gjm2TY
tweet.entities                {'hashtags': [{'text': 'NoPttGlobalCracker', 'indices': [38, 57]}, {'text': 'SavetheOhioRiver', 'in
dices': [112, 129]}, {'text': 'PlanetOrPlastic', 'indices': [130, 146]}, {'text': 'BreakFreeFromPlastic', 'indices': [147, 168]}],
'symbols': [], 'user_mentions': [{'screen_name': 'GCNewsl', 'name': 'GCNews', 'id': 1680442778, 'id_str': '1680442778', 'indices':
[59, 67]}, {'screen_name': 'GovMike...
tweet.user.id
71310291
tweet.user.screen_name
PlasticPollutes
tweet.user.followers_count
45953
tweet.user.friends_count
6595
tweet.user.favourites_count
13453
tweet.user.statuses_count
18732
tweet.retweet_count
4
tweet.favorite_count
12
tweet.favorited
False
tweet.retweeted
False
tweet.lang
en
fetchedAt
2019-12-29 07:02:29.624132
tweet.full_text_clean
Stop plastic pollution at its source.     protect our kids’ health and
anger
0.1169
fear
0.206122
joy
0.4375
love
0.0423767
```

```
sadness
0.211166
trust
0.0418169
identity_hate
0.00188033
insult
0.00336857
obscene
0.00359257
severe_toxic
0.00188861
threat
0.00197363
toxic
0.011558
stakeholder
NGO
Name: 50, dtype: object
```

In [7]:
```
ConversationsLean.sample(3)
```

Out[7]:

	Unnamed: 0	tweet.created_at	tweet.id	tweet.full_text	tweet.entities	tweet.user.id	tweet.user.screen_name	tweet.user.fo
147	101	2018-06-08 18:09:15.000000	1.005150e+18	Thanks for your answers everyone! We're excited to announce that you can now search for key research studies on plastic pollution through our new Science Research Hub at https://t.co/HozPXFP04M #sciencetosolutions #5gyres #breakfreefromplastic	{'hashtags': [{'text': 'sciencetosolutions', 'indices': [194, 213]}, {'text': '5gyres', 'indices': [214, 221]}, {'text': 'breakfreefromplastic', 'indices': [222, 243]}], 'symbols': [], 'user_mentions': [], 'urls': [{'url': 'https://t.co/HozPXFP04M', 'expanded_url': 'http://www.5gyres.org/science-hub', 'display_url': '5gyres.org/science-hub', 'indices': [170, 193]}]}	85732762	5gyres	

	Unnamed: 0	tweet.created_at	tweet.id	tweet.full_text	tweet.entities	tweet.user.id	tweet.user.screen_name	tweet.user.f
79	33	2019-04-14 00:05:04.000000	1.117217e+18	Last week 100 activists of the #breakfreefromplastic movement trooped to @Nestle Philippine headquarters to demand accountability for their role in abetting the country's plastic pollution crisis. #plasticpollutes \nhttps://t.co/iDWmfTmcxS https://t.co/MnafFyG8hW	{'hashtags': [{'text': 'breakfreefromplastic', 'indices': [31, 52]}, {'text': 'plasticpollutes', 'indices': [198, 214]}], 'symbols': [], 'user_mentions': [{'screen_name': 'Nestle', 'name': 'Nestlé', 'id': 23085995, 'id_str': '23085995', 'indices': [73, 80]}], 'urls': [{'url': 'https://t.co/iDWmfTmcxS', 'expanded_url': 'http://ow.ly/5QIm30opvjo', 'display_url': 'ow.ly/5QIm30opvjo', 'indices': [...	71310291	PlasticPollutes	
430	4	2017-12-28 00:00:27.000000	9.461689e+17	#DidYouKnow Americans alone discard 30+ mil tons of plastic a year, only 8% is recycled. #RefuseSingleUse plastic whenever possible 🌎	{'hashtags': [{'text': 'DidYouKnow', 'indices': [0, 11]}, {'text': 'RefuseSingleUse', 'indices': [89, 105]}], 'symbols': [], 'user_mentions': [], 'urls': []}	71310291	PlasticPollutes	

3 rows × 31 columns

In [8]:
```python
ConversationsLean.groupby('stakeholder').count()
```

Out[8]:

	Unnamed: 0	tweet.created_at	tweet.id	tweet.full_text	tweet.entities	tweet.user.id	tweet.user.screen_name	tweet.user.followers_count	twe
stakeholder									
Artist	104	104	104	104	104	104	104	104	
Multinational	20	20	20	20	20	20	20	20	
NGO	241	241	241	241	241	241	241	241	
OtherInstitution	51	51	51	51	51	51	51	51	

	Unnamed: 0	tweet.created_at	tweet.id	tweet.full_text	tweet.entities	tweet.user.id	tweet.user.screen_name	tweet.user.followers_count	twe
stakeholder									
Scientific	24	24	24	24	24	24	24	24	

5 rows × 30 columns

In [ ]:

Task 1. Which Twitter users are the most popular ? (10%)

For each Task please: (1) develop the code required and (2) provide a brief discussion and interpretation of the results

Tip: consider retweet and favorite counts as proxies for popularity

1a. Top 5 most popular users by number of Retweets.

In [465…

```
#Group by the user name the sum of the retweet counts
ret=ConversationsLean.groupby(['tweet.user.screen_name']).sum()[['tweet.retweet_count']]
ret.columns = ['Retweets']
ret.index.names = ['User']
#Sort values by retweets
ret_sorted=ret.sort_values(by='Retweets',ascending=False)
ret_sorted.head()
```

Out[465…

	Retweets
User	
PlasticPollutes	4242.0
Greenpeace	3941.0
5gyres	569.0

	Retweets
User	
Algalita	540.0
WRAP_UK	474.0

In [506…

```
#Create adataframe with top 5 users  by number of Retweets
ret_sorted_5= ret_sorted[:5]
ret_sorted_5
#Create a dataframe with the rest of users
ret_others_df= ret_sorted[5:]
#Sum all the Retweets other than the top 5
Other_retweets=ret_others_df['Retweets'].sum()
#Create a new row with the sum of all the Retweets other than the top 5
new_row = pd.DataFrame(data = {'Retweets' : [Other_retweets]},index=['Others'])
#Create a new dataframe with the top 5 users  by number of Retweets and the rest of users**
ret_final = pd.concat([ret_sorted_5, new_row])
ret_final.index.names = ['User']
ret_final
```

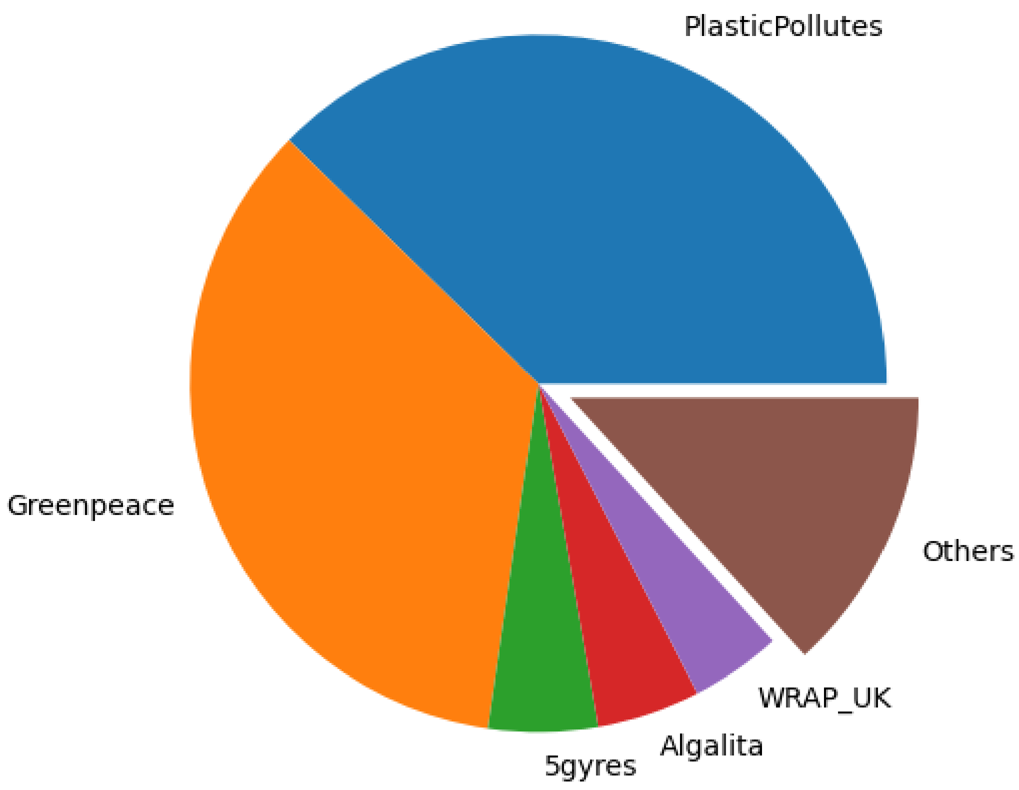
Out[506…

	Retweets
User	
PlasticPollutes	4242.0
Greenpeace	3941.0
5gyres	569.0
Algalita	540.0
WRAP_UK	474.0
Others	1489.0

In [464…

```
#Plot Top 5 most popular users by number of Retweets.
myexplode = [0, 0, 0, 0,0,.1]
ret_final.plot.pie(subplots=True, figsize=(16, 8), explode = myexplode,legend=None, fontsize=14)
plt.title('1a. Top 5 most popular users by number of Retweets.', fontsize=19,  color='red')
plt.ylabel(None)
plt.show()
```

1a. Top 5 most popular users by number of Retweets.



We can observe that PlasticPollutes followed by Greenpeace are the two users that has more Retweets, these users will be considered the mot popular users, followed by 5gyres, algalita and Wrap_UK.

1b. Top 5 most popular users by number of Favourites.

In [466…
```
#Group by the user name the sum of the Favourites given counts
ret2=ConversationsLean.groupby(['tweet.user.screen_name']).sum()[['tweet.favorite_count']]
```

```
ret2.columns = ['Favourites']
ret2.index.names = ['User']
#Sort values by retweets
ret_sorted2=ret2.sort_values(by='Favourites',ascending=False)
ret_sorted2.head()
```

Out[466…

	Favourites
User	
Greenpeace	6357.0
PlasticPollutes	4027.0
Nestle	688.0
5gyres	636.0
WRAP_UK	625.0

In [507…

```
#Create adataframe with top 5 users  by number of Favourites given
ret_sorted_5_2= ret_sorted2[:5]
#Create a dataframe with the rest of users
ret_others_df2= ret_sorted2[5:]
#Sum all the Favourites other than the top 5
Other_retweets2=ret_others_df2['Favourites'].sum()
#Create a new row with the sum of all the Favourites other than the top 5
new_row2 = pd.DataFrame(data = {'Favourites' : [Other_retweets]},index=['Others'])
#Create a new dataframe with the top 5 users  by number of Favourites and the rest of users**
ret_final2 = pd.concat([ret_sorted_5_2, new_row2])
ret_final2.index.names = ['User']
ret_final2
```

Out[507…

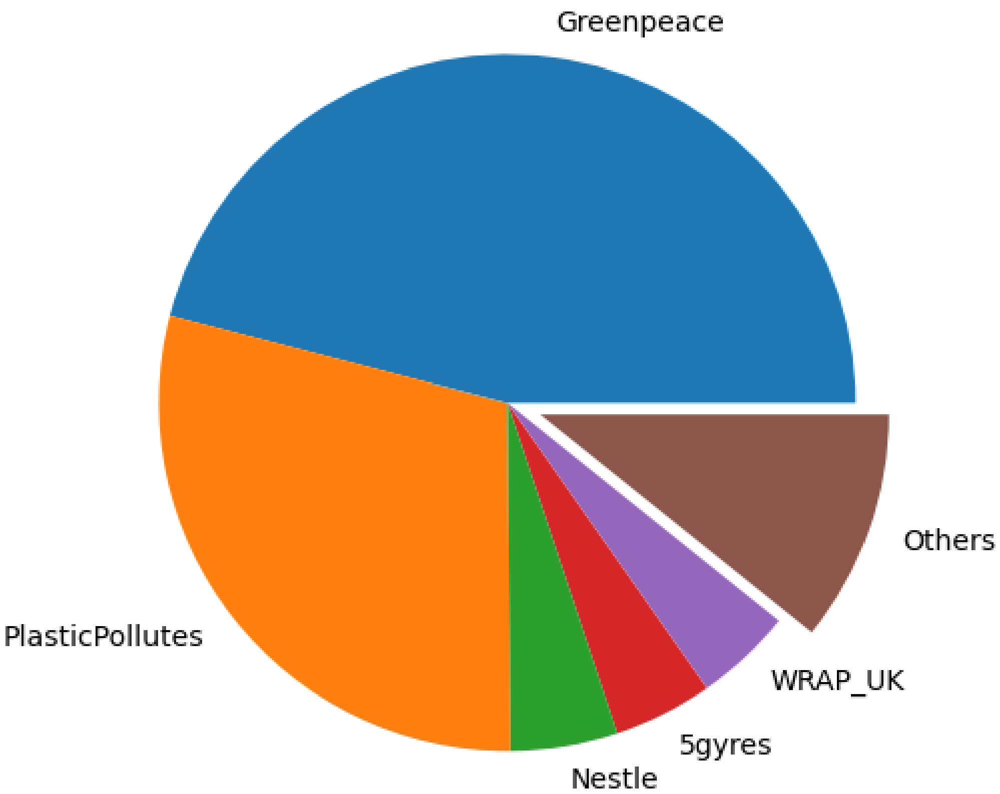
	Favourites
User	
Greenpeace	6357.0
PlasticPollutes	4027.0
Nestle	688.0
5gyres	636.0

	Favourites
User	
WRAP_UK	625.0
Others	1489.0

In [468…

```
#Plot Top 5 most popular users by number of Favourites.
myexplode = [0, 0, 0, 0,0,.1]
ret_final2.plot.pie(subplots=True, figsize=(16, 8), explode = myexplode,legend=None, fontsize=14)
plt.title('1b. Top 5 most popular users by number of Favourites.', fontsize=19,  color='red')
plt.ylabel(None)
plt.show()
```

1b. Top 5 most popular users by number of Favourites.



We can observe again that Greenpeace followed by PlasticPollutes are the two users that has more Favourites(likes), these users will be considered the mot popular users, followed by Nestle, 5gyres, and Wrap_UK.

1c. Calculating the weighted linear combination of retweets and favourite counts (50% -50%)

In order to have one list of popularity we can build a model of weighted linear combination of retweets and favourite counts.In this case, we gave equal weight to each element, but the marketing department staff can add some specific weights.

This calculation is achieved by normalizing both columns (forcing them to sum one); then calculte the sum of the columns multiplied by the chosen weight.

</h3>

In [12]:

```python
# Sum all the values for the coulumns retweet and favorite
sumfav=task1[['tweet.user.favourites_count']].sum()
sumret=task1[['tweet.retweet_count']].sum()
# Normalise each column dividing them by its sum
retweet_norm=task1['tweet.retweet_count'].apply(lambda x: x/sumret)
fav_norm=task1['tweet.user.favourites_count'].apply(lambda x: x/sumfav)
# Merge the columns into a new dataframe
df_merge_col = pd.merge(fav_norm, retweet_norm, on='tweet.user.screen_name')
#Calculate the weighted linear combination  of retweets and favourite counts (50%-50%)
df_merge_col['weighted_col']=df_merge_col['tweet.user.favourites_count']*.5+df_merge_col['tweet.retweet_count']*.5
#Sort the values
df_merge_col.sort_values(by='weighted_col',ascending=False).head(10)['weighted_col']
```

Out[12]:

```
tweet.user.screen_name
PlasticPollutes    0.512451
Greenpeace         0.211618
5gyres             0.047935
WRAP_UK            0.038181
Algalita           0.028710
Nestle             0.028471
NoPlasticStraws    0.026322
PlasticfreeBeth    0.024470
HealTheBay         0.021687
MaxLiboiron        0.012195
Name: weighted_col, dtype: float64
```

As in 1a and ib we can conclude that PlasticPollutes followed by Greenpeace are the most popular users taking in consideration the retweets made and favourites count, these companies together almost acomplish 3/4 of the total counts, followed by other well-known NGOs.

Task 2. Which Stakeholders users are the most emotional ? (10%)

For each Task please: (1) develop the code required and (2) provide a brief discussion and interpretation of the results

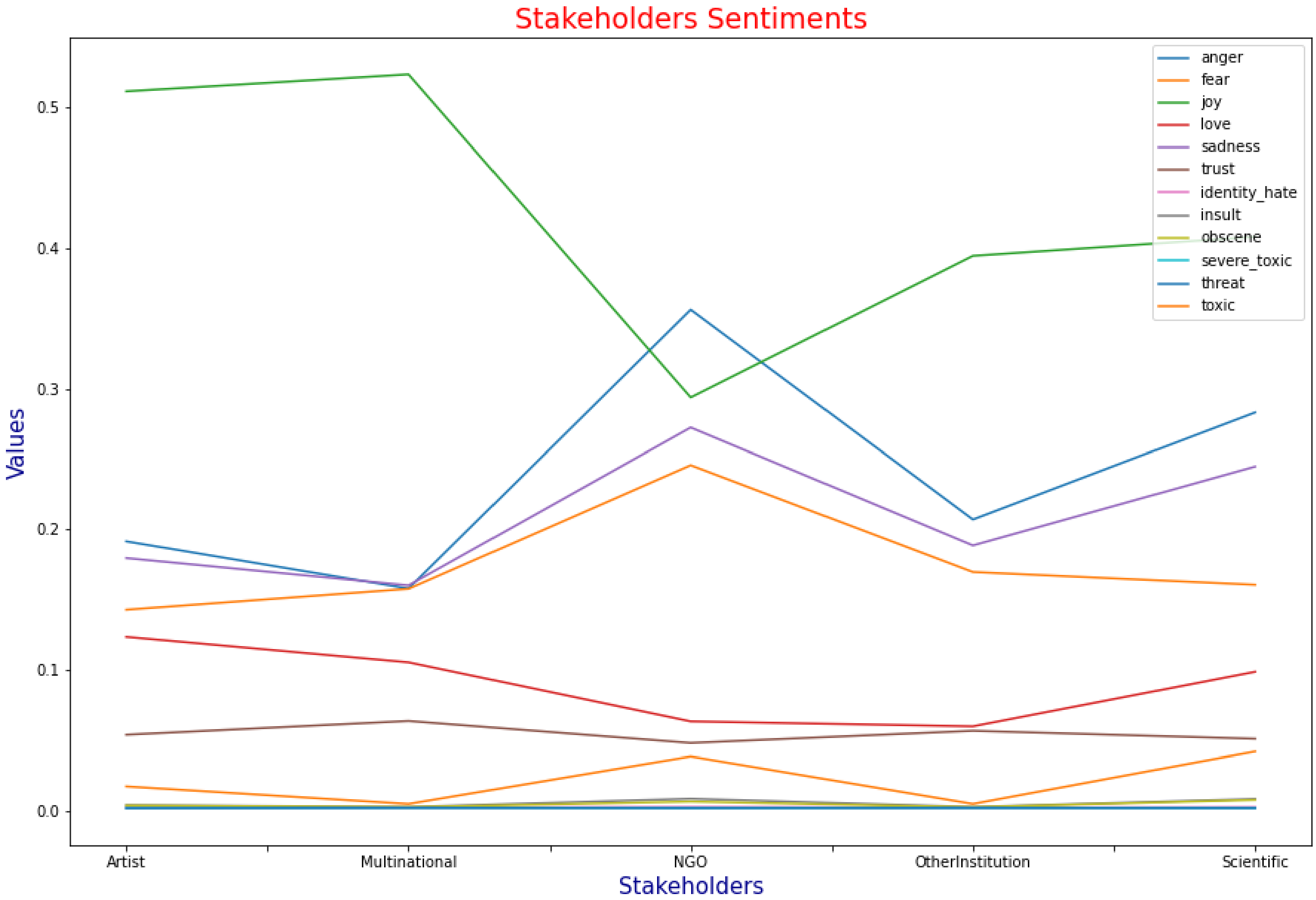
Tip: visualize levels of emotions accross stakeholders (NGOs, artists, Multinational)

In [13]:
```
# Group by stakeholders
task2=ConversationsLean.groupby(['stakeholder']).mean()[['anger','fear','joy','love','sadness','trust','identity_hate', 'insult',
task2
```

Out[13]:

stakeholder	anger	fear	joy	love	sadness	trust	identity_hate	insult	obscene	severe_toxic	threat	toxic
Artist	0.191531	0.142912	0.511195	0.123589	0.179675	0.054033	0.002312	0.004153	0.003898	0.001976	0.002006	0.017433
Multinational	0.158001	0.157787	0.523263	0.105483	0.160258	0.063770	0.002394	0.002865	0.002714	0.002283	0.002258	0.005048
NGO	0.356121	0.245599	0.293706	0.063486	0.272420	0.048228	0.002920	0.008715	0.006713	0.002021	0.002076	0.038602
OtherInstitution	0.207091	0.169765	0.394409	0.059978	0.188679	0.056795	0.002396	0.002867	0.002694	0.002226	0.002192	0.005053
Scientific	0.283057	0.160679	0.407954	0.098758	0.244664	0.051185	0.002766	0.008543	0.007881	0.001990	0.002107	0.042202

In [281…
```
# Plot emotions across Stakeholders
task2.plot(figsize=(15,10))
plt.title('Stakeholders Sentiments', fontsize=19,  color='red')
plt.xlabel('Stakeholders', fontsize=15, color='darkblue')
plt.ylabel('Values', fontsize=15, color='darkblue')
plt.show()
```



It can be concluded that non-governmental organizations have the strongest reactions towards negative emotions such as anger, sadness and fear; artists react more to positive emotions such as

love and joy. It is worth mentioning that scientists pparentely respond in a more balanced way between negative and positive emotions.

Task 3. Do emotions play a role in the number of retweets and favorites achieved by tweets ? (20 %)

For each Task please: (1) develop the code required and (2) provide a brief discussion and interpretation of the results

Tip: correlations between variables

Tip: visualizations relating variables

Tip: optionally consider basic regression models to determine the impact of some variables on others (e.g. impact of love on favorite, impact of fear on retweet)

3a. Create a dataframe with the required data

In [15]:
```
task3=ConversationsLean.groupby(['tweet.retweet_count','tweet.user.favourites_count']).mean()[['anger','fear','joy','love','sadnes
task3.columns = ['retweet', 'favourites','anger','fear','joy','love','sadness','trust','id_hate', 'insult', 'obscene', 'sevtoxic',
task3=task3.astype(float)
task3.sample(6)
```

Out[15]:

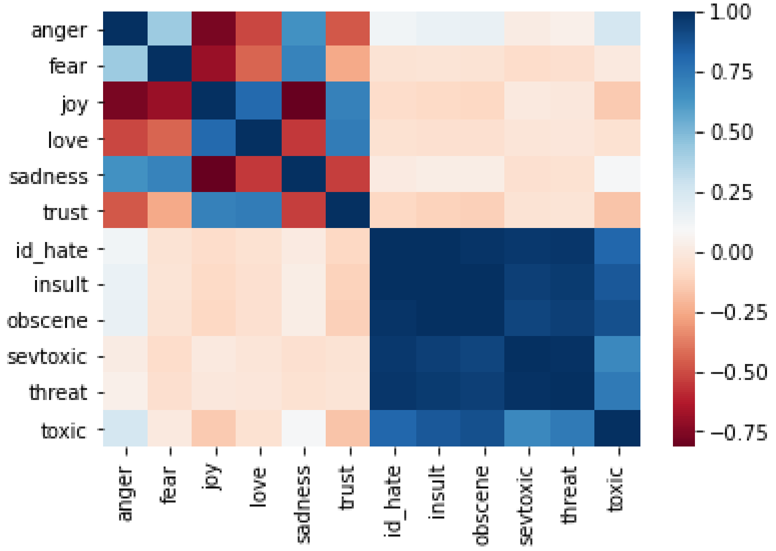
	retweet	favourites	anger	fear	joy	love	sadness	trust	id_hate	insult	obscene	sevtoxic	threat	toxic
174	65.0	1963.0	0.153798	0.112836	0.493568	0.072675	0.124153	0.083291	0.002333	0.002971	0.002728	0.002199	0.002262	0.004780
172	56.0	8752.0	0.274870	0.076770	0.320861	0.021720	0.109632	0.027815	0.002141	0.003875	0.003288	0.001495	0.001789	0.013885
135	24.0	13453.0	0.358063	0.305664	0.248876	0.032430	0.326943	0.042912	0.002428	0.005301	0.004606	0.001694	0.001793	0.030168
137	25.0	17795.0	0.808619	0.145627	0.047693	0.011549	0.253918	0.024276	0.012847	0.110629	0.101173	0.003432	0.004008	0.594284
192	123.0	13453.0	0.455632	0.114242	0.147103	0.020518	0.430449	0.026334	0.002306	0.004825	0.004645	0.001739	0.001891	0.026643
126	22.0	2025.0	0.699010	0.279167	0.067918	0.016587	0.278843	0.031708	0.002155	0.003106	0.002979	0.002002	0.002001	0.007840

3b. Check for Perfect Multicollinearity

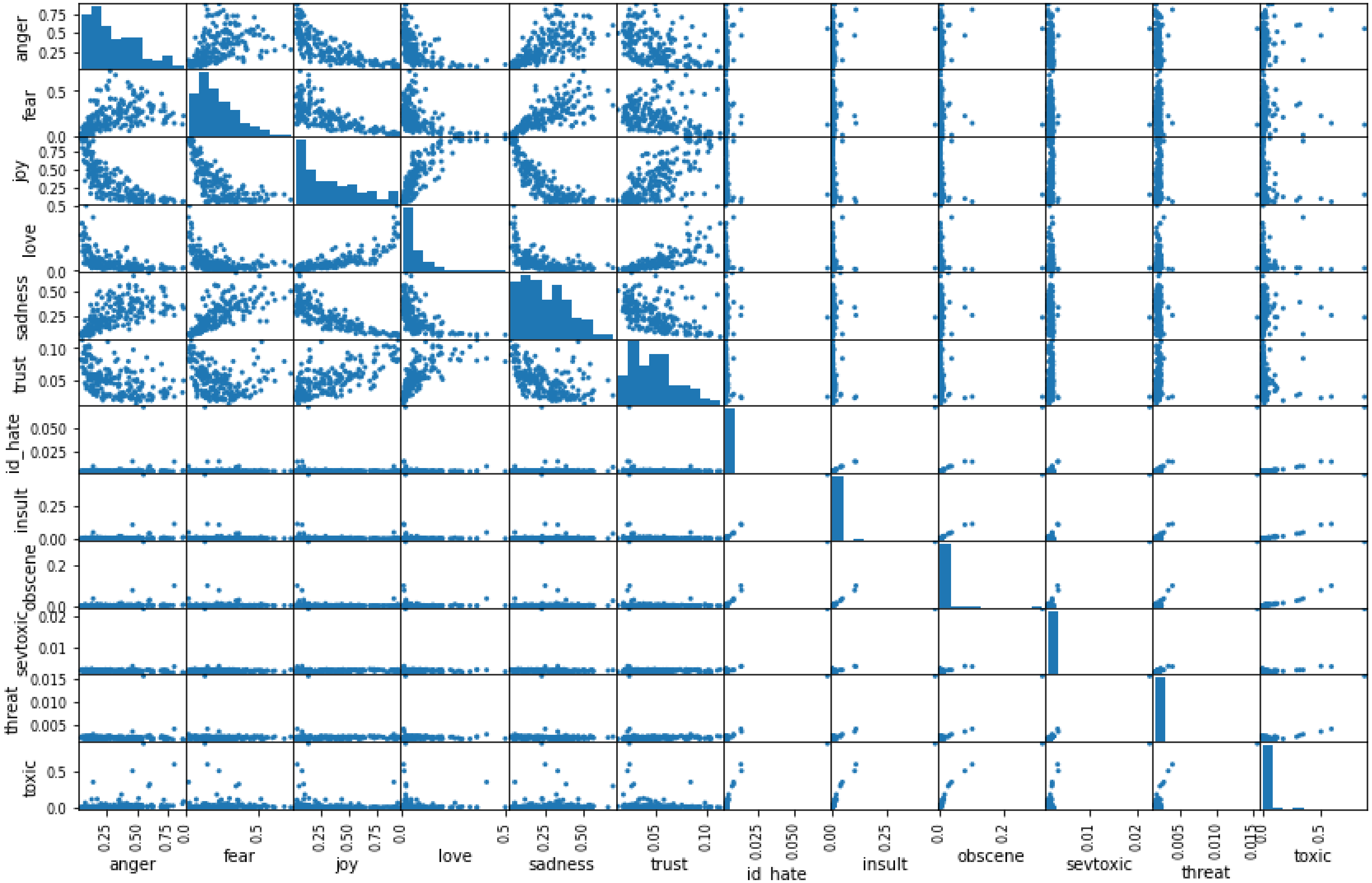
In [16]:
```
# calculate the correlation matrix  between variables
task3_a=task3.drop(['retweet','favourites'], axis = 1)
corr=task3_a.corr()
display(corr)
#Plot the correlation heatmap
sns.heatmap(corr, xticklabels=corr.columns, yticklabels=corr.columns, cmap='RdBu')
```

	anger	fear	joy	love	sadness	trust	id_hate	insult	obscene	sevtoxic	threat	toxic
anger	1.000000	0.425192	-0.770882	-0.511400	0.641449	-0.472639	0.124358	0.154396	0.163204	0.015757	0.038995	0.249127
fear	0.425192	1.000000	-0.691274	-0.432867	0.699025	-0.253874	-0.038243	-0.033545	-0.035474	-0.073258	-0.058415	0.005749
joy	-0.770882	-0.691274	1.000000	0.798126	-0.814269	0.708003	-0.071207	-0.091337	-0.093864	0.002533	-0.008775	-0.146349
love	-0.511400	-0.432867	0.798126	1.000000	-0.550945	0.724159	-0.043582	-0.051400	-0.050115	-0.020893	-0.015444	-0.046507
sadness	0.641449	0.699025	-0.814269	-0.550945	1.000000	-0.534175	0.010475	0.025391	0.026277	-0.054606	-0.045689	0.100516
trust	-0.472639	-0.253874	0.708003	0.724159	-0.534175	1.000000	-0.097871	-0.117107	-0.121350	-0.038451	-0.030817	-0.165192
id_hate	0.124358	-0.038243	-0.071207	-0.043582	0.010475	-0.097871	1.000000	0.994503	0.984536	0.968589	0.976610	0.815324
insult	0.154396	-0.033545	-0.091337	-0.051400	0.025391	-0.117107	0.994503	1.000000	0.996678	0.944209	0.959802	0.862186
obscene	0.163204	-0.035474	-0.093864	-0.050115	0.026277	-0.121350	0.984536	0.996678	1.000000	0.926519	0.948516	0.888329
sevtoxic	0.015757	-0.073258	0.002533	-0.020893	-0.054606	-0.038451	0.968589	0.944209	0.926519	1.000000	0.991864	0.680502
threat	0.038995	-0.058415	-0.008775	-0.015444	-0.045689	-0.030817	0.976610	0.959802	0.948516	0.991864	1.000000	0.735481
toxic	0.249127	0.005749	-0.146349	-0.046507	0.100516	-0.165192	0.815324	0.862186	0.888329	0.680502	0.735481	1.000000

Out[16]: <AxesSubplot:>



In [17]:
```python
#Plot the bivariate relationships between combinations of variables
pd.plotting.scatter_matrix(task3_a, alpha = 1, figsize = (14,9))
plt.show()
```



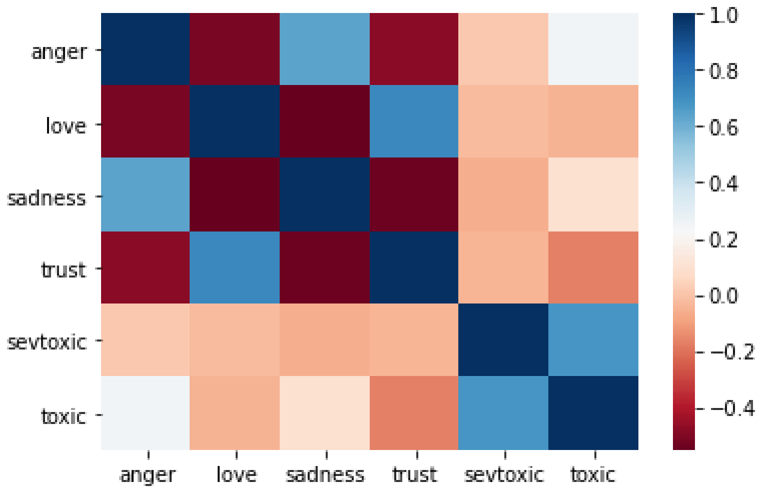
We can observe variables that are highly linearly related, we will drop that variables so the coefficients in our regression model are not artificially calculated.

In [18]:
```
# Eliminate the variables that are highly correlated
task3_drop = task3_a.drop(['id_hate','insult','threat','obscene',  'joy',  'fear'], axis = 1)
```

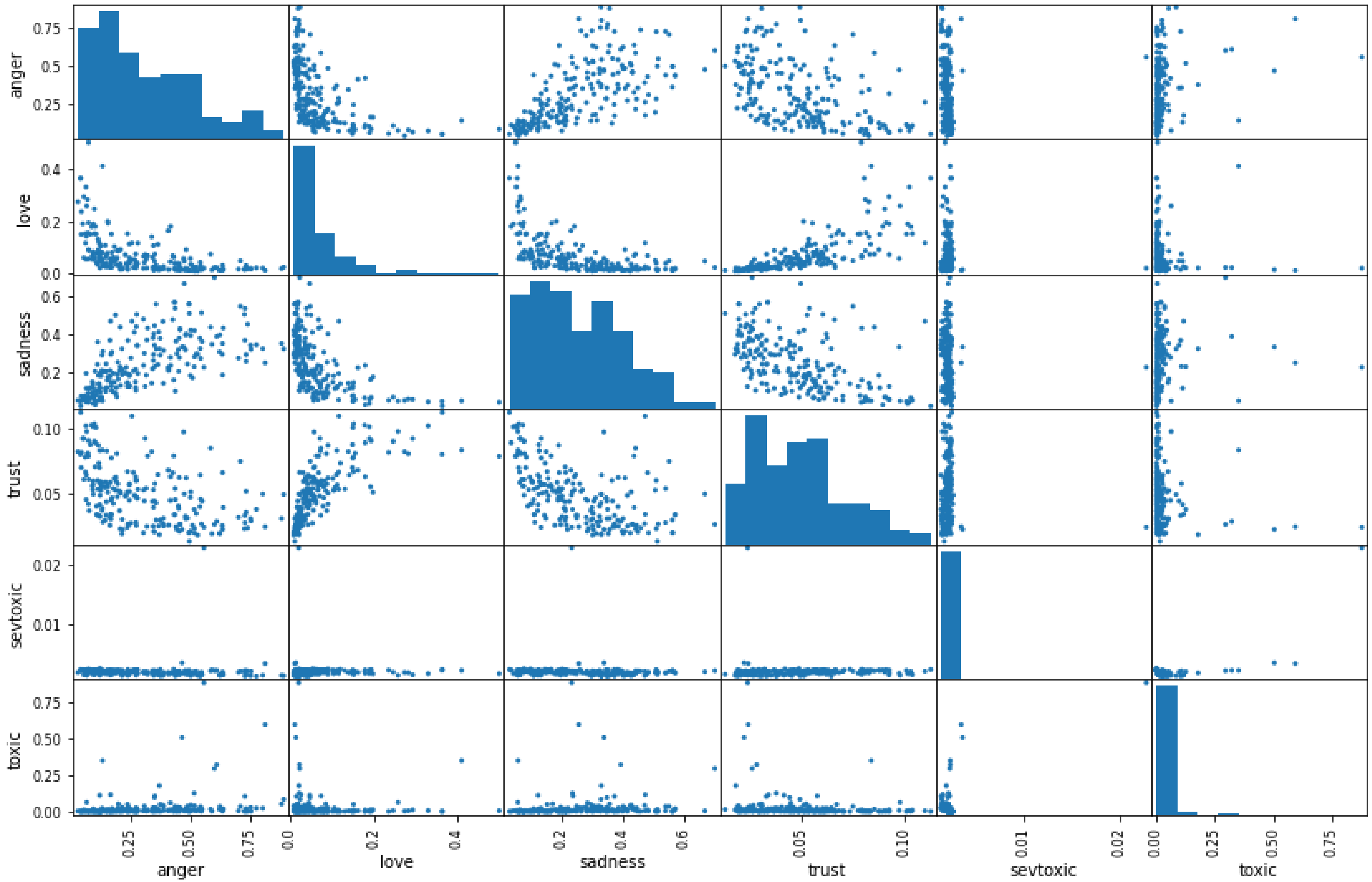
In [19]:
```
# calculate the correlation matrix  between variables
corr2=task3_drop.corr()
display(corr2)
#Plot correlation
sns.heatmap(corr2, xticklabels=corr2.columns, yticklabels=corr2.columns, cmap='RdBu')
```

	anger	love	sadness	trust	sevtoxic	toxic
anger	1.000000	-0.511400	0.641449	-0.472639	0.015757	0.249127
love	-0.511400	1.000000	-0.550945	0.724159	-0.020893	-0.046507
sadness	0.641449	-0.550945	1.000000	-0.534175	-0.054606	0.100516
trust	-0.472639	0.724159	-0.534175	1.000000	-0.038451	-0.165192
sevtoxic	0.015757	-0.020893	-0.054606	-0.038451	1.000000	0.680502
toxic	0.249127	-0.046507	0.100516	-0.165192	0.680502	1.000000

Out[19]: <AxesSubplot:>



In [20]:
```
#Plot the bivariate relationships between combinations of variables
pd.plotting.scatter_matrix(task3_drop, alpha = 1, figsize = (14,9))
plt.show()
```



Now we have the variables that are going to initially use our regression model. After that we will determine the level of importance of each predictor variable and eliminate the variables that are not significant to our regression model.

3c. Regression Model of retweets on emotions.

In [21]:
```
#Final Regression Model of Retweets on significant variables
X = task3[['joy', 'id_hate','insult','obscene']]
Y = task3['retweet']
model = sm.OLS(Y, X).fit()
print_model = model.summary()
print(print_model)
```

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                retweet   R-squared (uncentered):                   0.154
Model:                            OLS   Adj. R-squared (uncentered):              0.138
Method:                 Least Squares   F-statistic:                              9.457
Date:                Thu, 08 Apr 2021   Prob (F-statistic):                    4.83e-07
Time:                        18:43:45   Log-Likelihood:                         -1290.0
No. Observations:                 212   AIC:                                      2588.
Df Residuals:                     208   BIC:                                      2601.
Df Model:                           4
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
joy          -55.3481     29.314     -1.888      0.060    -113.138       2.442
id_hate     3.197e+04   6864.956      4.657      0.000    1.84e+04    4.55e+04
insult     -9833.0611   2807.986     -3.502      0.001   -1.54e+04   -4297.300
obscene     7432.4221   3683.649      2.018      0.045     170.349    1.47e+04
==============================================================================
Omnibus:                      302.024   Durbin-Watson:                   0.154
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            25618.674
Skew:                           6.475   Prob(JB):                         0.00
Kurtosis:                      55.273   Cond. No.                         407.
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

3d. Regression Model of favourites on emotions.

In [22]:
```
#Final Regression Model of favourites on significant variables
X1 = task3[['toxic','joy','insult','sevtoxic']]
Y1 = task3['favourites']
model1 = sm.OLS(Y1, X1).fit()
```

```
print_model1 = model1.summary()
print(print_model1)
```

```
                                 OLS Regression Results                                
=======================================================================================
Dep. Variable:             favourites   R-squared (uncentered):                   0.565
Model:                            OLS   Adj. R-squared (uncentered):              0.557
Method:                 Least Squares   F-statistic:                              67.52
Date:                Thu, 08 Apr 2021   Prob (F-statistic):                    1.54e-36
Time:                        18:43:45   Log-Likelihood:                         -2137.1
No. Observations:                 212   AIC:                                      4282.
Df Residuals:                     208   BIC:                                      4296.
Df Model:                           4                                                  
Covariance Type:            nonrobust                                                  
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
toxic       5.543e+04   8660.367      6.400      0.000    3.84e+04    7.25e+04
joy        -6753.6346   1612.322     -4.189      0.000   -9932.222   -3575.047
insult     -3.036e+05      3e+04    -10.105      0.000   -3.63e+05   -2.44e+05
sevtoxic    4.381e+06   3.77e+05     11.622      0.000    3.64e+06    5.12e+06
==============================================================================
Omnibus:                       11.253   Durbin-Watson:                   1.759
Prob(Omnibus):                  0.004   Jarque-Bera (JB):               12.114
Skew:                           0.584   Prob(JB):                      0.00234
Kurtosis:                       2.915   Cond. No.                         403.
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

We can conclude that emotions play an essential role in the number of retweets sent and the number of favourites (likes).
It is important to note that emotional influence is more significant in favourites counts, which we can observe in the adjusted square R is .557. We also note that negative emotions have a greater impact than positive emotions.

Task 4. Develop and apply a function which: (1) extracts ALL the hashtags from the column 'tweet.full_text', (2) saves the result as a new column (10%)

For each Task please: (1) develop the code required and (2) provide a brief discussion and interpretation of the results

Tip: consider lambda functions applied to a dataframe

Tip: refer to the module labs for examples on how to use REGEX in the context of pandas dataframes

In [755…
```
# Use a lambda function and REGEX to search patterns that contains the hashtags inthe column tweet.full_text
ConversationsLean['hashtags_extracted']=ConversationsLean['tweet.full_text'].apply(lambda x:re.findall(r'\B(\#[a-zA-Z]+\b)(?!;)',
```

In [756…
```
ConversationsLean['hashtags_extracted'].head(15)
```

Out[756…
```
0                          [#plasticpollutes, #recyclingisnottheanswer]
1          [#recycling, #plasticpollution, #plastics, #plasticpollutes]
2          [#recycling, #plasticpollution, #plastics, #plasticpollutes]
3                                                 [#reuse, #recycling]
4                       [#recyclingisnotenough, #breakfreefromplastic]
5     [#plastic, #recycling, #UKPlasticsPact, #tech, #CircularEconomy]
6                [#ukplasticspact, #changeplasticforgood, #recycling]
7                  [#plastic, #plastics, #recycling, #ukplasticspact]
8                      [#plastics, #changeplasticforgood, #recycling]
9                                               [#funding, #recycling]
10                                              [#funding, #recycling]
11                           [#recycling, #circulareconomy, #funding]
12                           [#recycling, #circulareconomy, #funding]
13                           [#recycling, #circulareconomy, #funding]
14                             [#funding, #smallbusiness, #recycling]
Name: hashtags_extracted, dtype: object
```

We can clearly observe the hashtags used to get the attention of twitter users. These hashtags are manly about plastic, the pollution it creates, sustainability and recycling.

Task 5. what are the differences between stakeholders ? (30%)

Tip: explore differences in terms of emotions, popularity, hashtags used, number of tweets,etc

5a. Differences in emotions across stakeholders.

As we already study emotions across stakeholders in Task 2, we will study negative and positive emotions across stakeholders.

5ai. Create a DataFrame of emotions across stakeholders.

In [199…

```
# Create a Dataframe with  emotions
emo=ConversationsLean.groupby(['stakeholder']).mean()[['anger','fear','joy','love','sadness','trust','identity_hate', 'insult', 'o
emo
```

Out[199…

stakeholder	anger	fear	joy	love	sadness	trust	identity_hate	insult	obscene	severe_toxic	threat	toxic
Artist	0.191531	0.142912	0.511195	0.123589	0.179675	0.054033	0.002312	0.004153	0.003898	0.001976	0.002006	0.017433
Multinational	0.158001	0.157787	0.523263	0.105483	0.160258	0.063770	0.002394	0.002865	0.002714	0.002283	0.002258	0.005048
NGO	0.356121	0.245599	0.293706	0.063486	0.272420	0.048228	0.002920	0.008715	0.006713	0.002021	0.002076	0.038602
OtherInstitution	0.207091	0.169765	0.394409	0.059978	0.188679	0.056795	0.002396	0.002867	0.002694	0.002226	0.002192	0.005053
Scientific	0.283057	0.160679	0.407954	0.098758	0.244664	0.051185	0.002766	0.008543	0.007881	0.001990	0.002107	0.042202

5aii. For a better visualisation we will normalise each column by diving it by its sum; then we will average all the positive feelings and all the negative feelings.

In [129…

```
# Sum all the values for each coulumns
sumjoy=emo[['joy']].sum();sumlove=emo[['love']].sum(); sumtrust=emo[['trust']].sum();
sumanger=emo[['anger']].sum();sumfear=emo[['fear']].sum();sumsadness=emo[['sadness']].sum(); sumidentity_hate=emo[['identity_hate'
```

```
suminsult=emo[['insult']].sum();sumobscene=emo[['obscene']].sum(); sumsevere_toxic=emo[['severe_toxic']].sum();
sumthreat=emo[['threat']].sum();sumtoxic=emo[['toxic']].sum()
```

In [130…

```
# Normalise each column dividing them by its sum
joy_n=emo['joy'].apply(lambda x: x/sumjoy);love_n=emo['love'].apply(lambda x: x/sumlove);trust_n=emo['trust'].apply(lambda x: x/su
anger_n=emo['anger'].apply(lambda x: x/sumanger);fear_n=emo['fear'].apply(lambda x: x/sumfear);sadness_n=emo['sadness'].apply(lamb
identity_hate_n=emo['identity_hate'].apply(lambda x: x/sumidentity_hate);insult_n=emo['insult'].apply(lambda x: x/suminsult);
obscene_n=emo['obscene'].apply(lambda x: x/sumobscene);severe_toxic_n=emo['severe_toxic'].apply(lambda x: x/sumsevere_toxic);
threat_n=emo['threat'].apply(lambda x: x/sumthreat);toxic_n=emo['toxic'].apply(lambda x: x/sumtoxic)
```

In [208…

```
# Merge all the normalised emotions into a  single dataframe
from functools import reduce
df_emo = [joy_n, love_n,trust_n,anger_n,fear_n,sadness_n,identity_hate_n,insult_n,obscene_n,severe_toxic_n,threat_n,toxic_n]
emo_n = reduce(lambda  left,right: pd.merge(left,right,on=['stakeholder'], how='outer'), df_emo)
emo_n
```

Out[208…

	joy	love	trust	anger	fear	sadness	identity_hate	insult	obscene	severe_toxic	threat	toxic
stakeholder												
Artist	0.239938	0.273856	0.197191	0.160169	0.163004	0.171823	0.180806	0.152998	0.163117	0.188282	0.188562	0.160910
Multinational	0.245603	0.233734	0.232728	0.132130	0.179970	0.153255	0.187228	0.105562	0.113551	0.217471	0.212269	0.046594
NGO	0.137856	0.140675	0.176008	0.297810	0.280127	0.260516	0.228356	0.321074	0.280872	0.192583	0.195150	0.356311
OtherInstitution	0.185123	0.132902	0.207273	0.173182	0.193631	0.180434	0.187329	0.105630	0.112720	0.212030	0.206004	0.046645
Scientific	0.191480	0.218834	0.186799	0.236709	0.183268	0.233973	0.216281	0.314736	0.329740	0.189635	0.198014	0.389540

In [203…

```
#Average of all positive normalised emotions, and average of all negative normalised emotions; add the averages to the dataframe
emo_n['positive_emotions']=(emo_n['joy']+emo_n['love']+emo_n['trust'])/3
emo_n['negative_emotions']=(emo_n['anger']+emo_n['fear']+emo_n['sadness']+emo_n['identity_hate']+emo_n['insult']+emo_n['obscene']+
emo_n
```

Out[203…

	joy	love	trust	anger	fear	sadness	identity_hate	insult	obscene	severe_toxic	threat	toxic	positive_en
stakeholder													
Artist	0.239938	0.273856	0.197191	0.160169	0.163004	0.171823	0.180806	0.152998	0.163117	0.188282	0.188562	0.160910	0.
Multinational	0.245603	0.233734	0.232728	0.132130	0.179970	0.153255	0.187228	0.105562	0.113551	0.217471	0.212269	0.046594	0.
NGO	0.137856	0.140675	0.176008	0.297810	0.280127	0.260516	0.228356	0.321074	0.280872	0.192583	0.195150	0.356311	0.

	joy	love	trust	anger	fear	sadness	identity_hate	insult	obscene	severe_toxic	threat	toxic	positive_en
stakeholder													
OtherInstitution	0.185123	0.132902	0.207273	0.173182	0.193631	0.180434	0.187329	0.105630	0.112720	0.212030	0.206004	0.046645	0.
Scientific	0.191480	0.218834	0.186799	0.236709	0.183268	0.233973	0.216281	0.314736	0.329740	0.189635	0.198014	0.389540	0.

In [204…

```
# Drop all unnecesary columns
emo_final=emo_n.drop(['anger','fear','joy','love','sadness','trust','identity_hate', 'insult', 'obscene', 'severe_toxic', 'threat'
emo_final
```

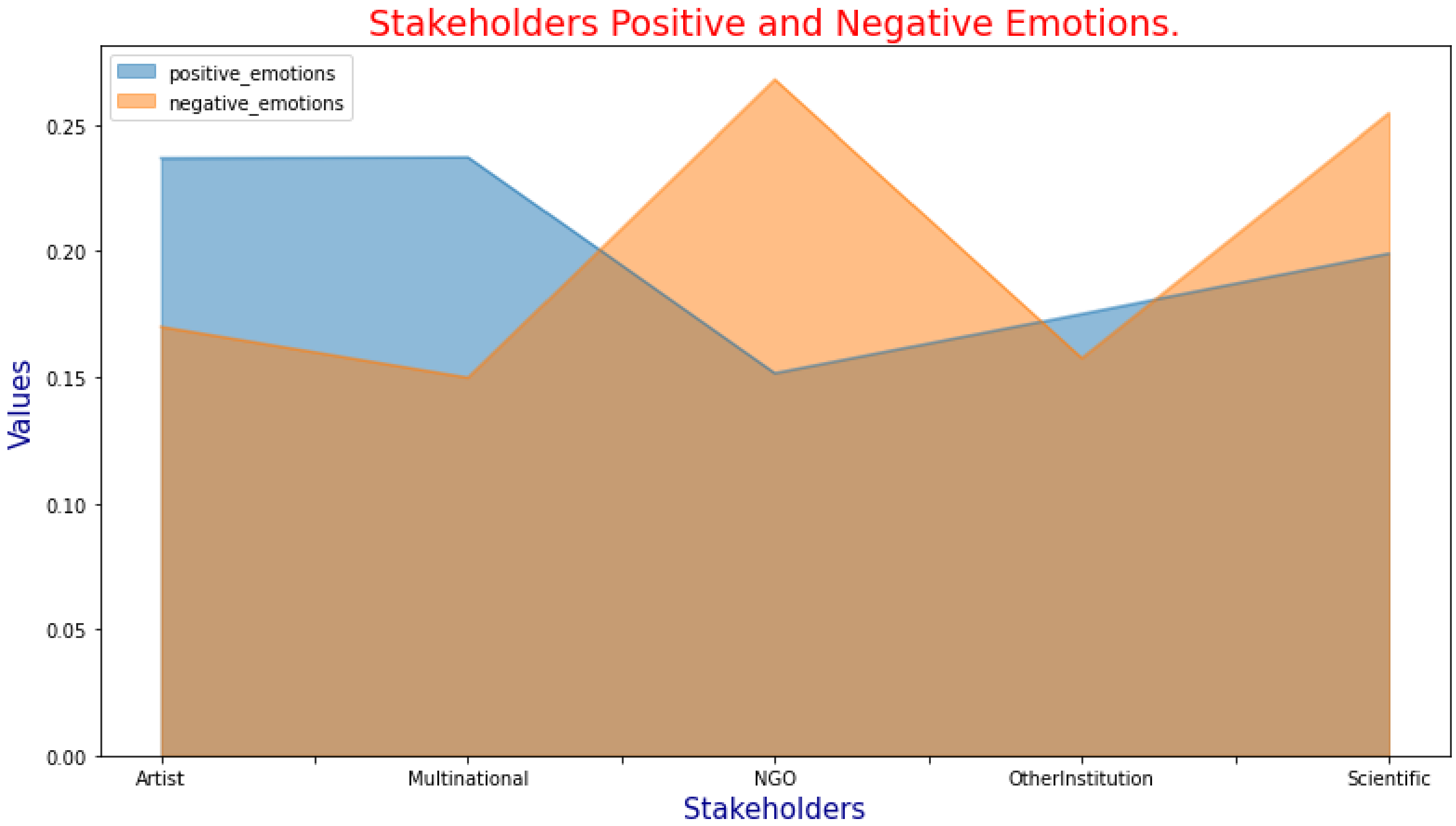
Out[204…

	positive_emotions	negative_emotions
stakeholder		
Artist	0.236995	0.169963
Multinational	0.237355	0.149781
NGO	0.151513	0.268089
OtherInstitution	0.175099	0.157512
Scientific	0.199038	0.254655

5aiii. Plot positive and the negative feelings accross stakeholders.

In [519…

```
# Plot negative and positive emotions across Stakeholders
emo_final.plot.area(figsize=(13,7), stacked=False)
plt.title('Stakeholders Positive and Negative Emotions.', fontsize=19,  color='red')
plt.xlabel('Stakeholders', fontsize=15, color='darkblue')
plt.ylabel('Values', fontsize=15, color='darkblue')
plt.show()
```



We can observe that non-governmental organizations have the strongest reactions towards negative emotions; artists and multinationals react more to positive emotions; other institutions have similar reactions to negative and positive emotions; Scientifics reacts more to negative than positive emotions.

5b. Differences in popularity across stakeholders.

In [221…

```
#Create a Dataframe of the number of followers and friends across stakeholders.
popularity=ConversationsLean.groupby(['stakeholder']).mean()[['tweet.user.followers_count','tweet.user.friends_count']].sort_value
```

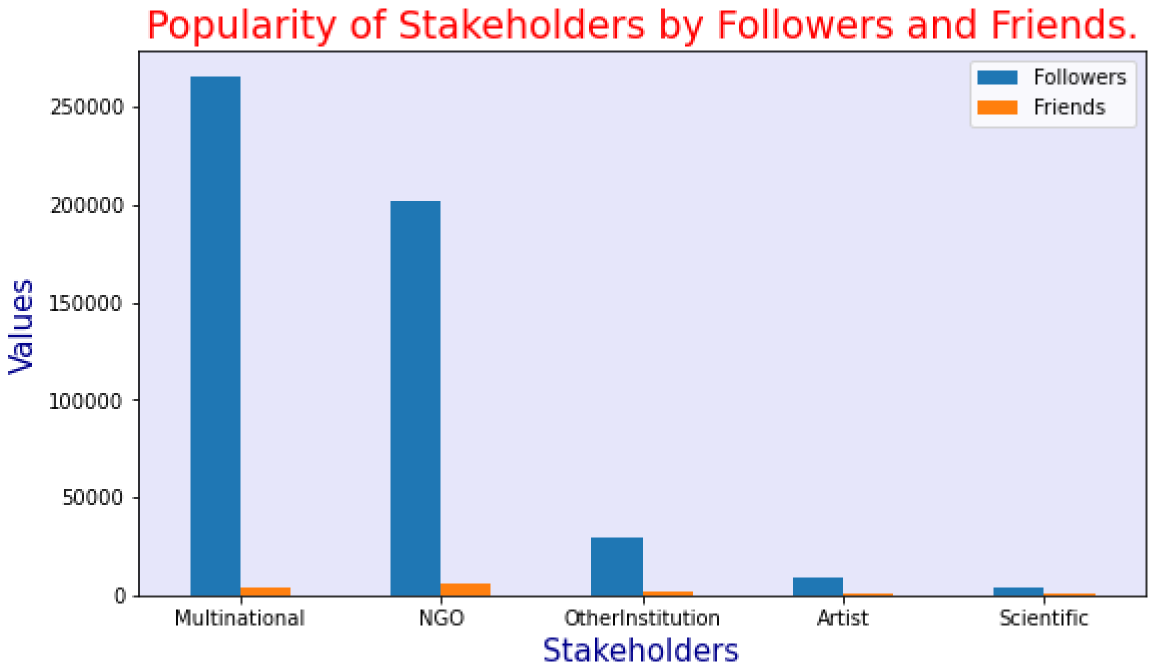
```
popularity.columns = ['Followers', 'Friends']
popularity
```

Out[221…

	Followers	Friends
stakeholder		
Multinational	265708.0	3855.0
NGO	202046.0	5557.0
OtherInstitution	29593.0	1723.0
Artist	9121.0	1180.0
Scientific	4179.0	923.0

In [283…

```
# Plot  popularity across stakeholders measured  in followers and friends.
popularity.plot.bar(figsize=(9,5),  stacked=False).set_facecolor("lavender")
plt.title('Popularity of Stakeholders by Followers and Friends.', fontsize=19,  color='red')
plt.xlabel('Stakeholders', fontsize=15, color='darkblue')
plt.ylabel('Values', fontsize=15, color='darkblue')
plt.xticks(rotation=0)
plt.show()
```



We can observe that multinationals are the organizations followed by more people; in second place, non-governmental organizations have a considerable amount of followers. Regarding friends, the differences are not substantial; being Non-governmental, the stakeholder following more people.

In [279…

```
#Create a Dataframe of the number of retweets and likes across stakeholders.
popularity2=ConversationsLean.groupby(['stakeholder']).sum()[['tweet.favorite_count','tweet.retweet_count']].sort_values(by='tweet
popularity2.columns = ['Retweets', 'Favourites']
popularity2
```

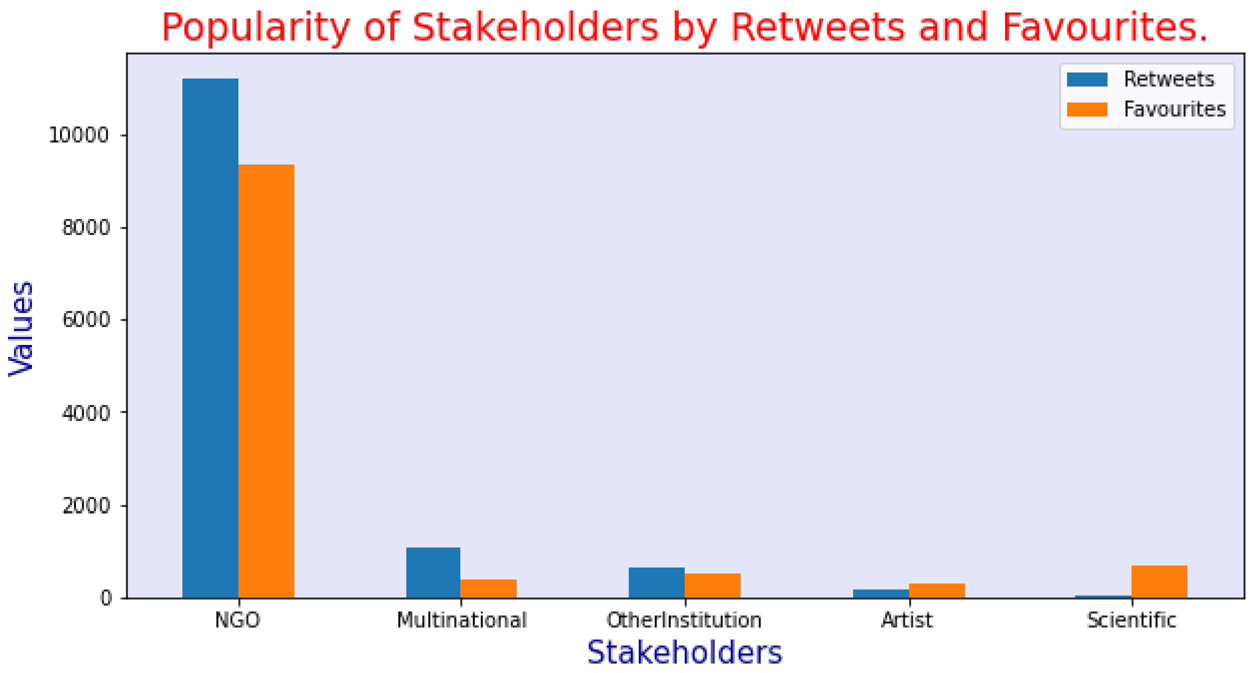
Out[279…

	Retweets	Favourites
stakeholder		
NGO	11196.0	9322.0
Multinational	1057.0	391.0
OtherInstitution	633.0	512.0

	Retweets	Favourites
stakeholder		
Artist	146.0	281.0
Scientific	30.0	680.0

In [284…

```
# Plot  popularity across stakeholders measured  in retweets and likes.
popularity2.plot.bar(figsize=(10,5),  stacked=False).set_facecolor("lavender")
plt.title('Popularity of Stakeholders by Retweets and Favourites.', fontsize=19,  color='red')
plt.xlabel('Stakeholders', fontsize=15, color='darkblue')
plt.ylabel('Values', fontsize=15, color='darkblue')
plt.xticks(rotation=0)
plt.show()
```



We can affirm that non-governmental organizations without a doubt have the greater impact with their communication in twitter.

5c. Activity (number of tweets and likes) of stakeholders.

In [521...

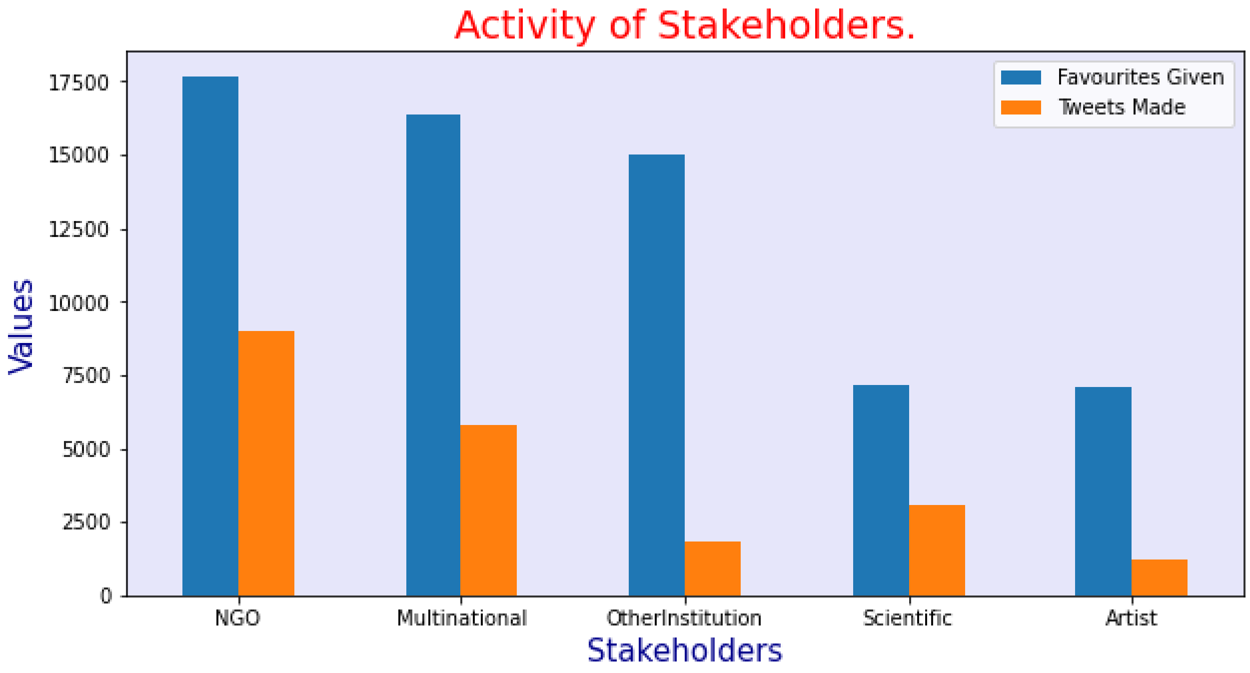
```
#Create a Dataframe of the number of tweets and likes made by stakeholders
activity=ConversationsLean.groupby(['stakeholder']).mean()[['tweet.user.statuses_count','tweet.user.favourites_count']].sort_value
activity.columns = ['Favourites Given', 'Tweets Made']
activity
```

Out[521...

stakeholder	Favourites Given	Tweets Made
NGO	17671.0	9000.0
Multinational	16380.0	5810.0
OtherInstitution	15012.0	1802.0
Scientific	7152.0	3046.0
Artist	7086.0	1204.0

In [285...

```
# Plot the number of tweets and likes made by stakeholders
activity.plot.bar(figsize=(10,5),  stacked=False).set_facecolor("lavender")
plt.title('Activity of Stakeholders.', fontsize=19,  color='red')
plt.xlabel('Stakeholders', fontsize=15, color='darkblue')
plt.ylabel('Values', fontsize=15, color='darkblue')
plt.xticks(rotation=0)
plt.show()
```



In this comparison we can see how active the organizations are on twitter. This graph reflects the commitment they have with their communications on social networks and as in the comparison of popularity we can see that civil society organizations are in first place, followed by multinationals and other institutions. It is important to note that the difference is not significant as in other areas.

5d. Differences in hashtags across stakeholders.

5di. Most used hashtags in NGOs.

In [770…

```
#Create a Dataframe with stakeholders and hastags.
hash=ConversationsLean[['stakeholder','hashtags_extracted']]
#Make a Filter just to have just NGOs
NGO_filter=hash['stakeholder']=='NGO'
hash_NGO_serie=hash[NGO_filter]
#Put a hashtag per row
```

```
NGO = pd.Series([item for sublist in hash_NGO_serie.hashtags_extracted for item in sublist])
#Convert to lowercase the hashtags
NGO = NGO.astype(str).str.lower()
#Count the frequency of avery hastag
df_NGO1 = NGO.value_counts()
df_NGO1.head(5)
```

Out[770…

```
#breakfreefromplastic    112
#plasticpollutes          97
#plasticfree              36
#plasticispoison          26
#plastickills             23
dtype: int64
```

In [801…

```
#Generate a WordCloud Plot
wordcloud.generate_from_frequencies(df_NGO1)
fig = plt.figure(
    figsize = (8, 30),
    facecolor = 'k',
    edgecolor = 'k')
plt.imshow(wordcloud, interpolation = 'bilinear')
plt.axis('off')
plt.tight_layout(pad=0)
plt.title('NGOs  Hashtags', fontsize=42,  color='YELLOW')
plt.show()
```

We can observe the most used hashtags applied by non-governmental organizations are #breakfreefromplastic followed by #plasticpollutes.

5dii. Most used hashtags in Multinationals.

In [786…

```
#Make a Filter just to have just Multinationals
M_filter=hash['stakeholder']=='Multinational'
hash_M_serie=hash[M_filter]
#Put a hashtag per row
M = pd.Series([item for sublist in hash_M_serie.hashtags_extracted for item in sublist])
#Convert to lowercase the hashtags
M = M.astype(str).str.lower()
#Count the frequency of avery hastag
df_M1 = M.value_counts()
df_M1.head(5)
```

Out[786…

```
#circulareconomy          14
```

```
#sustainability          6
#beatplasticpollution    6
#recycling               5
#didyouknow              3
dtype: int64
```

In [802…

```
#Generate a WordCloud plot to visualise most used hashtags
wordcloud.generate_from_frequencies(df_M1)
fig = plt.figure(
    figsize = (8, 30),
    facecolor = 'k',
    edgecolor = 'k')
plt.imshow(wordcloud, interpolation = 'bilinear')
plt.axis('off')
plt.tight_layout(pad=0)
plt.title('Multinational  Hashtags', fontsize=42,  color='YELLOW')
plt.show()
```

We can observe the most used hashtags applied by Multinationals are #circulareconomy followed by #sustainability.

5diii. Most used hashtags in Other Institutions.

In [803…

```
#Make a Filter just to have just Multinationals
O_filter=hash['stakeholder']=='OtherInstitution'
hash_O_serie=hash[O_filter]
#Put a hashtag per row
O = pd.Series([item for sublist in hash_O_serie.hashtags_extracted for item in sublist])
#Convert to lowercase the hashtags
O = O.astype(str).str.lower()
#Count the frequency of avery hastag
df_O1 = O.value_counts()
df_O1.head(5)
```

Out[803…

```
#circulareconomy    30
#recycling          26
#plasticfree        16
#funding            10
#ukplasticspact      8
dtype: int64
```

In [805…

```
#Generate a WordCloud plot to visualise most used hashtags
wordcloud.generate_from_frequencies(df_O1)
fig = plt.figure(
    figsize = (8, 30),
    facecolor = 'k',
    edgecolor = 'k')
plt.imshow(wordcloud, interpolation = 'bilinear')
plt.axis('off')
plt.tight_layout(pad=0)
plt.title('Other Institution Hashtags', fontsize=42,  color='YELLOW')
plt.show()
```

We can observe the most used hashtags applied by Other Institutions are #circulareconomy followed by #recycling.

5div. Most used hashtags in Scientific Institutions.

In [807…

```python
#Make a Filter just to have just Multinationals
S_filter=hash['stakeholder']=='Scientific'
hash_S_serie=hash[S_filter]
#Put a hashtag per row
S = pd.Series([item for sublist in hash_S_serie.hashtags_extracted for item in sublist])
#Convert to lowercase the hashtags
S = S.astype(str).str.lower()
#Count the frequency of avery hastag
df_S1 = S.value_counts()
df_S1.head(5)
```

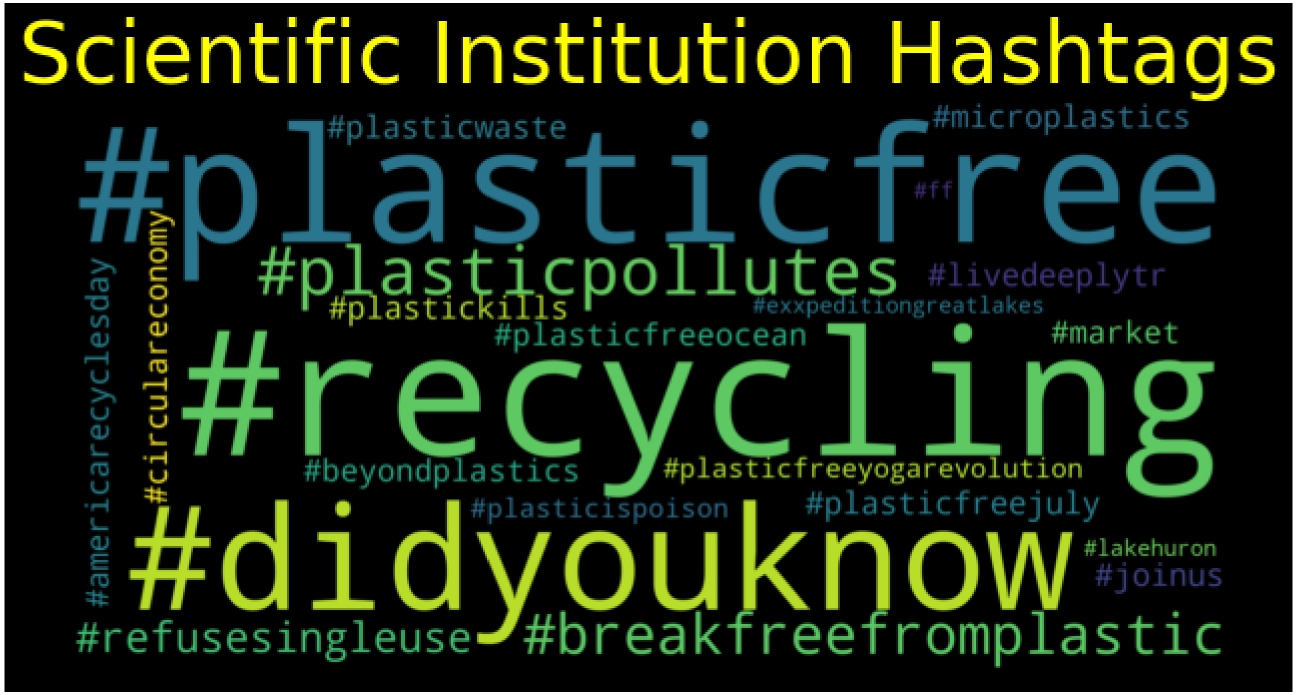
Out[807…

```
#recycling          6
```

```
#plasticfree            5
#didyouknow             4
#plasticpollutes        3
#breakfreefromplastic   2
dtype: int64
```

In [810…

```
#Generate a WordCloud plot to visualise most used hashtags
wordcloud.generate_from_frequencies(df_S1)
fig = plt.figure(
    figsize = (8, 30),
    facecolor = 'k',
    edgecolor = 'k')
plt.imshow(wordcloud, interpolation = 'bilinear')
plt.axis('off')
plt.tight_layout(pad=0)
plt.title('Scientific Institution Hashtags', fontsize=42,  color='YELLOW')
plt.show()
```



We can observe the most used hashtags applied by Scientific Institutions are #recycling followed by #plasticfree.

5dv. Most used hashtags by Artists.

In [814…

```
#Make a Filter just to have just Multinationals
A_filter=hash['stakeholder']=='Artist'
hash_A_serie=hash[M_filter]
#Put a hashtag per row
A = pd.Series([item for sublist in hash_A_serie.hashtags_extracted for item in sublist])
#Convert to lowercase the hashtags
A = A.astype(str).str.lower()
#Count the frequency of avery hastag
df_A1 = A.value_counts()
df_A1.head(5)
```

Out[814…

```
#plasticfreebeth    26
#plasticfree        17
#ecowed              4
#wastedialog         3
#plasticpollutes     2
dtype: int64
```

In [815…

```
#Generate a WordCloud plot to visualise most used hashtags
wordcloud.generate_from_frequencies(df_A1)
fig = plt.figure(
    figsize = (8, 30),
    facecolor = 'k',
    edgecolor = 'k')
plt.imshow(wordcloud, interpolation = 'bilinear')
plt.axis('off')
plt.tight_layout(pad=0)
plt.title('ArtistS Hashtags', fontsize=42,  color='YELLOW')
plt.show()
```

We can observe the most used hashtags applied by Scientific Institutions are #plasticfreebeth followed by #plasticfree .

Thanks to the hashtag visualizations, we can observe the different positions the different stakeholders have. For example, non-governmental organizations and artists are focused on eliminating plastic and raising awareness of the damage caused to the environment; Scientists are focused on eliminating and recycling plastic products; at the other extreme, we have multinationals where other interests are observed, like circular economy sustainability and recycling.

Task 6. what are your recommendations for Coca Cola as far as social media is concerned ? (20%)

This study analyses Twitter campaigns' negative impact due to Coca Cola plastic's use. This study was centred on the popularity, analysis of emotions, activity and focus of the communications of the different stakeholders involved. In summary:

- The most popular Twitter users are non-governmental organisations. Just Plastic Pollution Coalition and Greenpeace have approximately 75% of the retweets and favourites, followed by other governmental organisations. It is important to note that Nestle has a significant share in the favourites count.

- The analysis of emotions shows that non-governmental organisations' communications tend to have negative emotions such as anger, sadness and fear. At the same time, stakeholders like Artist focus on positive feelings in their communications.

- Regarding popularity, multinationals and non-governmental organisations have achieved successful campaigns having a significant number of Followers. It is crucial noting that this popularity is not reflected in the impact of tweets. The number of retweets and favourites is significantly higher in non-governmental organisations.

- The context of stakeholder communications can be analysed by the hashtags used. The hashtags related to the elimination of plastic and awareness of the damage caused to the environment prevail in stakeholders such as non-governmental organisations and artists; Scientists have a significant focus on recycling; multinationals try to focus on issues such as the circular economy, sustainability and recycling.

In conclusion, negative feelings towards Coca Cola which deteriorates the brand's concept, must be changed. Therefore, this study suggests three main strategies that Coca Cola should follow: More commitment to Coca Colas communications on social media regarding plastic pollution. The only presence that Coca Cola has is negative mentions; there is no involvement in these campaigns. Communication on Twitter is essential to change consumers' perception. Campaigns should focus on Circular Economy, promoting a recycling culture with a positive and optimistic approach to contrast the current negative emotions.

Shift the attention of non-governmental organisations and the general public to other industries. Dumped fishing gear is the most severe plastic pollution in the ocean. The most prominent fishing industry participants created the non-profit organisation called Marine Stewardship Council, which has a labelling fishery certification program to recognise sustainable fishing practices; this certification is just a trick to cover up their predatory fishing practices, and the damage they do to the environment. Similarly, a non-profit organisation can be created to focus on plastic pollution that fish consumption creates, generating a labelling certification that recognises recycling programs. This non-profit organisation should have substantial social networks activity with high use of negative emotions, currently used hashtags against plastic pollution and recycling.

Finally, and the most critical strategy is to create an authentic environmental culture in Coca Cola. Creating a genuine interest in plastic pollution could solve the root problem. Through research and technology, Coca Cola could produce plastics with biodegradable polymers at a lower cost;

offering an alternative to plastic such as glass or aluminium will create a commitment between the user and the company in favour of the environment.