Online Policing

Importing Modules

In [1]:
```
import re
import math
import json
import tweepy
import pandas as pd
import plotly.express as px

from keys import *
from tqdm import tqdm
from datetime import datetime
from collections import Counter
```

Authentication

In [2]:
```
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_secret)
api = tweepy.API(auth, wait_on_rate_limit=True, wait_on_rate_limit_notify=True)
setup = False
```

Data Collection

Police Account

In [3]:
```
username = '@MumbaiPolice'
user = api.get_user(id=username)
ID = user._json['id']
print('Handle:'+username)
print('ID:'+str(user._json['id']))
print('Name: '+user._json['name'])
print('Verified: '+str(user._json['verified']))
print('Account Creation: '+user._json['created_at'])
```

```
Handle:@MumbaiPolice
ID:4573405572
Name: Mumbai Police
Verified: True
Account Creation: Wed Dec 16 08:29:03 +0000 2015
```

Collecting Tweets

In [4]:
```
if setup:

    police_tweets = []

    for tweet in tqdm(tweepy.Cursor(api.user_timeline,user_id=ID,tweet_mode = "extended").items()):
        police_tweets.append(tweet._json)

    json.dump(police_tweets,open('./Dump/'+username[1:]+'_police_tweets.json','w'))
```

Collecting Tweets to which Police replied

In [5]:
```
if setup:

    police_tweets = json.load(open('./Dump/'+username[1:]+'_police_tweets.json','r'))
    candidates = set()
    people_tweets = []

    for tweet in tqdm(police_tweets):
        if tweet['in_reply_to_status_id']!=None and tweet['in_reply_to_user_id']!=tweet['user']['id']:
            candidates.add(tweet['in_reply_to_status_id'])

    print(len(candidates))
```

In [6]:
```
if setup:

    for i in tqdm(candidates):
        try:
            x = api.get_status(i,tweet_mode = "extended")._json
            if 'MumbaiPolice' in [i['screen_name'] for i in x['entities']['user_mentions']]:
                people_tweets.append(x)
        except:
            pass

    json.dump(people_tweets,open('./Dump/'+username[1:]+'_people_tweets.json','w'))
```

Analysis

Loading Collected Data

In [7]:
```
police_tweets = json.load(open('./Dump/'+username[1:]+'_police_tweets.json','r'))
people_tweets = json.load(open('./Dump/'+username[1:]+'_people_tweets.json','r'))
print('Total Tweets:',len(police_tweets))
print('Total Replies:',len(people_tweets))
```

```
Total Tweets: 3250
Total Replies: 2458
```

Regex

In [8]:
```
phone = '([+][9][1]|[9][1]|[0]){0,1}\s*([6-9]{1})\s*([0-9]{9})\s'
email = '[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,}'
aadhar = '[0-9]{4}\s[0-9]{4}\s[0-9]{4}'
pan = '[A-Z]{5}[0-9]{4}[A-Z]{1}'
passport = '\s[A-Z]{1}[0-9]{7}\s'
vehicle = '[A-Z]{2}\s[0-9]{2}\s[A-Z]{2}\s[0-9]{4}'
```

Collecting PII

In [9]:
```
PII = {'Phone':[],'Email':[],'Aadhar':[],'PAN':[],'Passport':[],'Vehicle':[]}
PII_Tweets = {'Phone': set(),'Email':set(),'Aadhar':set(),'PAN':set(),'Passport':set(),'Vehicle':set()}
S = set()

for tweet in people_tweets:
    tid = tweet['id']
    text = tweet['full_text']

    for x in re.findall(phone,text):
        PII['Phone'].append(''.join(x))
        PII_Tweets['Phone'].add(tid)
        S.add(tid)

    for x in re.findall(email,text):
        PII['Email'].append(x)
        PII_Tweets['Email'].add(tid)
        S.add(tid)

    for x in re.findall(aadhar,text):
        PII['Aadhar'].append(x)
        PII_Tweets['Aadhar'].add(tid)
        S.add(tid)

    for x in re.findall(pan,text):
        PII['PAN'].append(x)
        PII_Tweets['PAN'].add(tid)
        S.add(tid)

    for x in re.findall(passport,text):
        PII['Passport'].append(x)
        PII_Tweets['Passport'].add(tid)
        S.add(tid)

    for x in re.findall(vehicle,text):
        PII['Vehicle'].append(x)
        PII_Tweets['Vehicle'].add(tid)
        S.add(tid)
```

In [10]:
```
print('Number of Tweets with at least one PII:',len(S))
print()
print('Number of PIIs Detected')
print()
for i in PII:
    print(i+":"+" "*(8-len(i)),len(PII[i]))
print()

print('Tweets with PII in Text')
print()
for i in PII_Tweets:
    print(i+":",(PII_Tweets[i]))
    print()
```

```
Number of Tweets with at least one PII: 99

Number of PIIs Detected

Phone:    92
Email:    0
Aadhar:   0
PAN:      0
Passport: 0
Vehicle:  22

Tweets with PII in Text

Phone: {1374070773539737600, 1373334410385231873, 1379040873992781826, 1383359553820000258, 137765671
2602128385, 1382177101495738369, 1371654642908549126, 1379065067459670016, 1378967362116853764, 13826
22614128975878, 1383101078036647937, 1375463374075486211, 1376032924747718660, 1371785581743673351, 1
380479715652198410, 1382616877327781894, 1375391496342437891, 1375828010197127177, 137764814256953344
9, 1372558646664601611, 1374379198475694091, 1382331504462807044, 1370964197518020608, 13801805050316
75904, 1372108199155830784, 1373354660308815873, 1374281571767783425, 1380607880190038017, 1383833742
934904834, 1378976251449200642, 1379053498994593795, 1383058891622076419, 1371055134638182403, 138188
7891878514693, 1379057798219108363, 1372225025353211904, 1371367036626952192, 1381163823390842880, 13
78582957930110976, 1380793997183000576, 1374267653263564801, 1383057631611265025, 137107134644523008
0, 1371314481096138752, 1374611750088187904, 1375727532859777026, 1372072469998149634, 13746042212243
94754, 1370240533352701954, 1382704903974445058, 1374769510050242563, 1372149797415657475, 1371734733
416112131, 1374659626730364931, 1380865396429873155, 1378653619247931395, 1370997798741549061, 138076
1191409098757, 1375169005200154629, 1372206790834724871, 1381294158032240640, 1372047841867264000, 13
80893286643068928, 1372424201064214528, 1372610960490328064, 1370850824553787393, 137245447843194060
9, 1380770937675476993, 1379048330110496769, 1371389523280556034, 1381655807754792963, 13720834598295
92067, 1383075522389766148, 1379015785088118788, 1376348028966039556, 1374661882510929924, 1379399227
176939532}

Email: set()

Aadhar: set()

PAN: set()

Passport: set()

Vehicle: {1372865650251558912, 1383009011973251072, 1375856270566064130, 1374586880382005250, 1374272
301483446276, 1376935499995045892, 1370326492836626437, 1376907338834178048, 1375284772885291016, 137
6066378432532482, 1373936685629669378, 1370264793064513540, 1371450305766825996, 1375353408543096833,
1373925504399638530, 1375334751867871233, 1372928374407008263, 1375841314890149892, 13729065314321940
48, 1370766714833543168, 1380625418248581121, 1379775635485204480}
```

Tweets with Media

In [11]:
```
media_tweets = []
for tweet in people_tweets:
    if 'media' in tweet['entities']:
        media_tweets.append(tweet['id'])
print("Number of Tweets with Media:",len(media_tweets))
```

```
Number of Tweets with Media: 1264
```

Response Time

In [12]:
```
Reply = []
Reply_Time = []

Complaint = []
Complaint_Time = []

for x in people_tweets:
    for y in police_tweets:
        if y['in_reply_to_status_id'] == x['id']:
            Complaint.append(x['id'])
            Complaint_Time.append(datetime.strptime(x['created_at'],'%a %b %d %H:%M:%S +0000 %Y'))
            Reply.append(y['id'])
            Reply_Time.append(datetime.strptime(y['created_at'],'%a %b %d %H:%M:%S +0000 %Y'))
```

In [13]:
```
df = pd.DataFrame()
df['Complaint'] = Complaint
df['Reply'] = Reply
df['Complaint_Time'] = Complaint_Time
df['Reply_Time'] = Reply_Time
df['Response_Time'] = [Reply_Time[i]-Complaint_Time[i] for i in range(len(df))]
df.sort_values(by='Response_Time',inplace=True,ignore_index=True)
df
```

Out[13]:

	Complaint	Reply	Complaint_Time	Reply_Time	Response_Time
0	1383000601341009922	1383000658337431552	2021-04-16 10:13:32	2021-04-16 10:13:46	0 days 00:00:14
1	1381102962773139457	1381103031442280451	2021-04-11 04:33:00	2021-04-11 04:33:17	0 days 00:00:17
2	1378691728874475521	1378691818229964808	2021-04-04 12:51:37	2021-04-04 12:51:58	0 days 00:00:21
3	1384010967915261956	1384011092125356032	2021-04-19 05:08:23	2021-04-19 05:08:52	0 days 00:00:29
4	1370634746343002113	1370634891382026245	2021-03-13 07:16:03	2021-03-13 07:16:37	0 days 00:00:34
...	...	...	...	...	...
2466	1382047103044575234	1382393003021537287	2021-04-13 19:04:41	2021-04-14 17:59:10	0 days 22:54:29
2467	1381901946018095107	1382324583433334787	2021-04-13 09:27:53	2021-04-14 13:27:17	1 days 03:59:24
2468	1379722639590055941	1380327271093821440	2021-04-07 09:08:05	2021-04-09 01:10:41	1 days 16:02:36
2469	1376752720539918337	1377504215338127361	2021-03-30 04:26:42	2021-04-01 06:12:52	2 days 01:46:10
2470	1380114933761798145	1381096641638240257	2021-04-08 11:06:56	2021-04-11 04:07:53	2 days 17:00:57

2471 rows × 5 columns

Response Time Statistics

In [14]:
```
print("Min:",df['Response_Time'].min().round('1s'))
print("Max:",df['Response_Time'].max().round('1s'))
print("Mean:",df['Response_Time'].mean().round('1s'))
print("Standard Deviation:",df['Response_Time'].std().round('1s'))
```

```
Min: 0 days 00:00:14
Max: 2 days 17:00:57
Mean: 0 days 00:24:02
Standard Deviation: 0 days 02:18:54
```

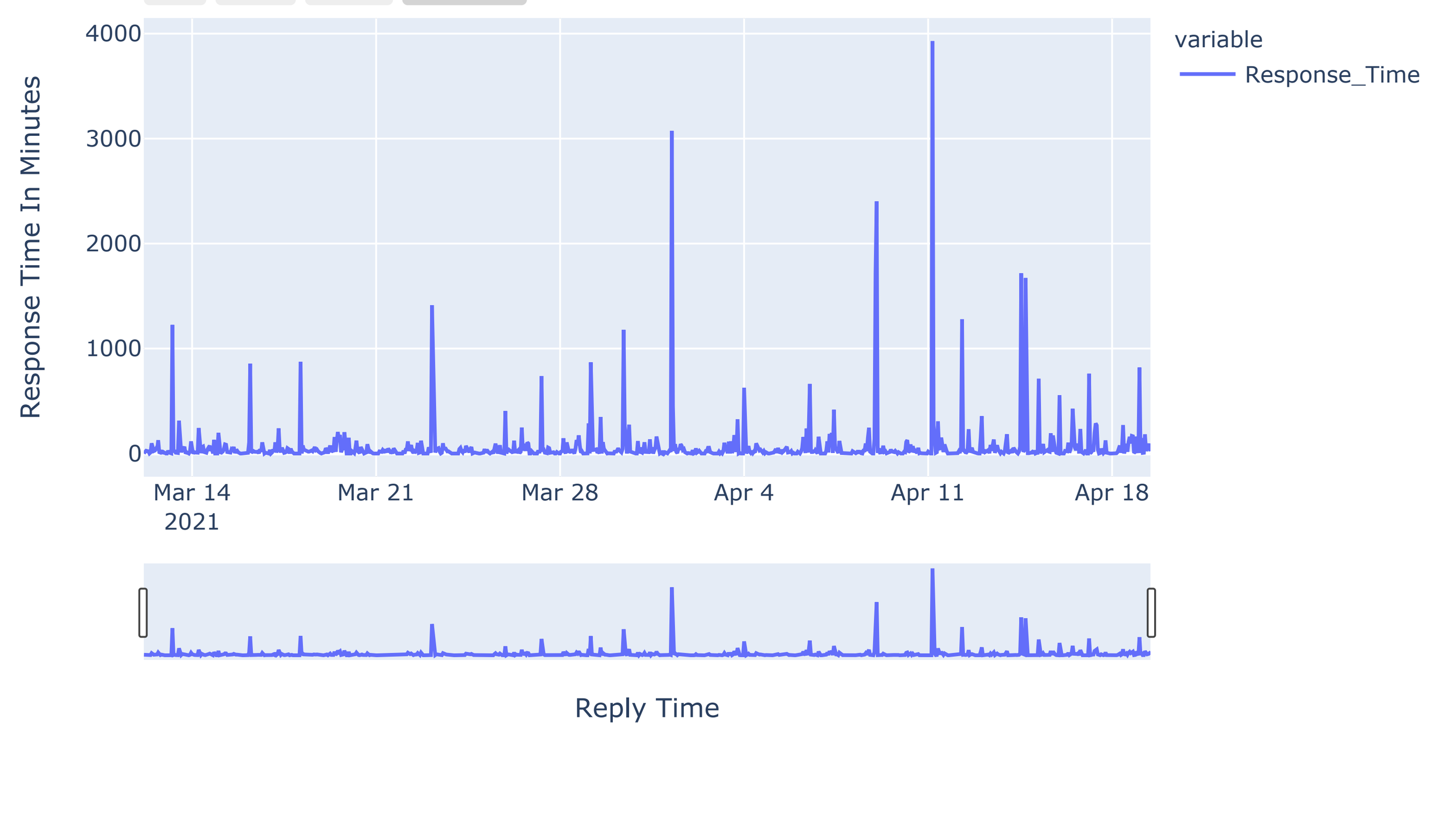
Time Series Plot of Response Time

In [15]:
```
Buttons = []
Buttons.append(dict(count=1, label="Day", step="day", stepmode="backward"))
Buttons.append(dict(count=7, label="Week", step="day", stepmode="backward"))
Buttons.append(dict(count=1, label="Month", step="month", stepmode="backward"))
Buttons.append(dict(label="Complete", step="all"))
```

In [16]:
```
s = 'H'
response = pd.DataFrame()
response['Reply_Time']=df['Reply_Time']
response['Response_Time']=[i.total_seconds()/60 for i in df['Response_Time']]
response = response.sort_values('Reply_Time')
response = response.resample(s, on='Reply_Time').Response_Time.sum()
response
```

Out[16]:
```
Reply_Time
2021-03-12 04:00:00    23.333333
2021-03-12 05:00:00    14.666667
2021-03-12 06:00:00    26.583333
2021-03-12 07:00:00    23.066667
2021-03-12 08:00:00    20.850000
                         ...
2021-04-19 07:00:00    27.450000
2021-04-19 08:00:00    42.950000
2021-04-19 09:00:00    56.866667
2021-04-19 10:00:00    93.316667
2021-04-19 11:00:00    21.733333
Freq: H, Name: Response_Time, Length: 920, dtype: float64
```

In [17]:
```
fig = px.line(response, title='Variation of Response Time of '+user._json['name'])
fig.update_xaxes(rangeslider_visible=True, rangeselector = dict(buttons=Buttons),title={'text':'Reply T
ime'})
fig.update_yaxes(title={'text':'Response Time In Minutes'})
fig.show()
```



Tweet IDs

In [18]:
```
with open('section_I.txt','w') as file:
    for tweet in police_tweets:
        file.write(str(tweet['id'])+'\n')
```