Reference:

- [https://www.youtube.com/watch?v=eFdPGpny_hY&t=98s (https://www.youtube.com/watch?v=eFdPGpny_hY&t=98s)](https://www.youtube.com/watch?v=eFdPGpny_hY&t=98s)
- Added some notes and made some changes

In [1]:
```python
import sys,tweepy,csv,re
#the following for stentiment analysis
from textblob import TextBlob
import matplotlib.pyplot as plt

```

In [3]:

```python
class SentimentAnalysis:

    def __init__(self):
        self.tweets = []
        self.tweetText = []

    def DownloadData(self):
        # authenticating
        access_token = '1219125691028930560-CZyXhFlgCpMM8rG11KwuYJaMoX7u
        access_secret = '0qSAxEoLmH9pKDPVFy2pQzbloaGxRcbb0JJgnkhq5F2d4'
        consumer_key = 'HpU6B5BVTuwAfa5nYX1vAVxgD'
        consumer_secret = 'Cgcs5YIHpIp5Pu5US3N0XAX8N4jlJgPQrE4aK8LYkM89r

        auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
        auth.set_access_token(access_token, access_secret)
        api = tweepy.API(auth)

        # input for term to be searched and how many tweets to search
        searchTerm = input("Enter Keyword/Tag to search about: ")
        NoOfTerms = int(input("Enter how many tweets to search: "))

        # searching for tweets
        self.tweets = tweepy.Cursor(api.search, q=searchTerm, lang = "er

        # Open/create a file to append data to
        csvFile = open('result.csv', 'a')

        # Use csv writer
        csvWriter = csv.writer(csvFile)


        # creating some variables to store info
        polarity = 0
        positive = 0
        wpositive = 0
        spositive = 0
        negative = 0
        wnegative = 0
        snegative = 0
        neutral = 0

         # iterating through tweets fetched
        for tweet in self.tweets:
            #Append to temp so that we can store in csv later. I use enc
            self.tweetText.append(self.cleanTweet(tweet.text).encode('ut
            # print (tweet.text.translate(non_bmp_map))    #print tweet'
            analysis = TextBlob(tweet.text)
            # print(analysis.sentiment)  # print tweet's polarity
            polarity += analysis.sentiment.polarity  # adding up polarit

            if (analysis.sentiment.polarity == 0):  # adding reaction of
                neutral += 1
            elif (analysis.sentiment.polarity > 0 and analysis.sentiment
                wpositive += 1
            elif (analysis.sentiment.polarity > 0.3 and analysis.sentime
                positive += 1
```

```python
57              elif (analysis.sentiment.polarity > 0.6 and analysis.sentime
58                  spositive += 1
59              elif (analysis.sentiment.polarity > -0.3 and analysis.sentim
60                  wnegative += 1
61              elif (analysis.sentiment.polarity > -0.6 and analysis.sentim
62                  negative += 1
63              elif (analysis.sentiment.polarity > -1 and analysis.sentimer
64                  snegative += 1
65          # Write to csv and close csv file
66          csvWriter.writerow(self.tweetText)
67          csvFile.close()
68
69          # finding average of how people are reacting
70          positive = self.percentage(positive, NoOfTerms)
71          wpositive = self.percentage(wpositive, NoOfTerms)
72          spositive = self.percentage(spositive, NoOfTerms)
73          negative = self.percentage(negative, NoOfTerms)
74          wnegative = self.percentage(wnegative, NoOfTerms)
75          snegative = self.percentage(snegative, NoOfTerms)
76          neutral = self.percentage(neutral, NoOfTerms)
77
78          # finding average reaction
79          polarity = polarity / NoOfTerms
80
81          # printing out data
82          print("How people are reacting on " + searchTerm + " by analyzir
83          print()
84          print("General Report: ")
85
86          if (polarity == 0):
87              print("Neutral")
88          elif (polarity > 0 and polarity <= 0.3):
89              print("Weakly Positive")
90          elif (polarity > 0.3 and polarity <= 0.6):
91              print("Positive")
92          elif (polarity > 0.6 and polarity <= 1):
93              print("Strongly Positive")
94          elif (polarity > -0.3 and polarity <= 0):
95              print("Weakly Negative")
96          elif (polarity > -0.6 and polarity <= -0.3):
97              print("Negative")
98          elif (polarity > -1 and polarity <= -0.6):
99              print("Strongly Negative")
100
101         print()
102         print("Detailed Report: ")
103         print(str(positive) + "% people thought it was positive")
104         print(str(wpositive) + "% people thought it was weakly positive"
105         print(str(spositive) + "% people thought it was strongly positiv
106         print(str(negative) + "% people thought it was negative")
107         print(str(wnegative) + "% people thought it was weakly negative"
108         print(str(snegative) + "% people thought it was strongly negativ
109         print(str(neutral) + "% people thought it was neutral")
110
111         self.plotPieChart(positive, wpositive, spositive, negative, wneg
112
113     def cleanTweet(self, tweet):
```

```python
114            # Remove Links, Special Characters etc from tweet
115            return ' '.join(re.sub("(@[A-Za-z0-9]+)|([^0-9A-Za-z \t]) | (\w
116
117        # function to calculate percentage
118        def percentage(self, part, whole):
119            temp = 100 * float(part) / float(whole)
120            return format(temp, '.2f')
121
122        def plotPieChart(self, positive, wpositive, spositive, negative, wne
123            labels = ['Positive [' + str(positive) + '%]', 'Weakly Positive
124                      'Negative [' + str(negative) + '%]', 'Weakly Negative
125            sizes = [positive, wpositive, spositive, neutral, negative, wneg
126            colors = ['yellowgreen','lightgreen','darkgreen', 'gold', 'red',
127            patches, texts = plt.pie(sizes, colors=colors, startangle=90)
128            plt.legend(patches, labels, loc="best")
129            plt.title('How people are reacting on ' + searchTerm + ' by anal
130            plt.axis('equal')
131            plt.tight_layout()
132            plt.show()
133
134  if __name__ == "__main__":
135        sa = SentimentAnalysis()
136        sa.DownloadData()
137
```

```
Enter Keyword/Tag to search about: mother's day
Enter how many tweets to search: 100
How people are reacting on mother's day by analyzing 100 tweets.

General Report:
Weakly Positive

Detailed Report:
25.00% people thought it was positive
13.00% people thought it was weakly positive
20.00% people thought it was strongly positive
2.00% people thought it was negative
8.00% people thought it was weakly negative
1.00% people thought it was strongly negative
31.00% people thought it was neutral
```
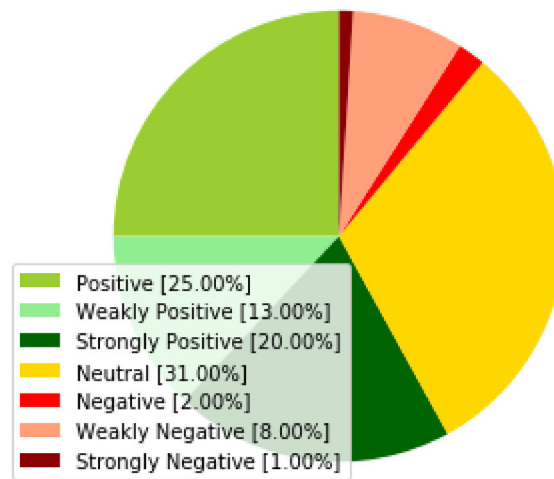
How people are reacting on mother's day by analyzing 100 Tweets.

Positive [25.00%]
Weakly Positive [13.00%]
Strongly Positive [20.00%]
Neutral [31.00%]
Negative [2.00%]
Weakly Negative [8.00%]
Strongly Negative [1.00%]

In [ ]: ▶| 1