Query based Sentiment Analysis using Python

In [1]:

```
# General:
import tweepy           # To consume Twitter's API
import pandas as pd     # To handle data
import numpy as np      # For number computing

# For plotting and visualization:
from IPython.display import display
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Creating a Twitter App

In [2]:

```
# Twitter App access keys for @user

# Consume:
CONSUMER_KEY    = 'yzxNyCyjkVJJrpNY0L3AphSVV'
CONSUMER_SECRET = 'e7xhbuIqTM520PCtwGrSrSIXWyrQbH7ckr9s0ToOlQE6tek4OP'

# Access:
ACCESS_TOKEN  = '2186056028-BPexxHqPrML7hcvwIuLfX3itD8IZPqguuksDHtn'
ACCESS_SECRET = 'vvhbgIqa7jK5nBAO9P4Mh1MTFvJdWH3Fl5bbzGhai9Dah'
```

In [3]:

```
# We import our access keys:
from credentials import *    # This will allow us to use the keys as variables

# API's setup:
def twitter_setup():
    """
    Utility function to setup the Twitter's API
    with our access keys provided.
    """
    # Authentication and access using keys:
    auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
    auth.set_access_token(ACCESS_TOKEN, ACCESS_SECRET)

    # Return API with authentication:
    api = tweepy.API(auth)
    return api
```

In [4]:

```
extractor = twitter_setup()
trends1 = extractor.trends_place(1) # from the end of your code
# trends1 is a list with only one element in it, which is a
# dict which we'll put in data.
data = trends1[0]
# grab the trends
trends = data['trends']
# grab the name from each trend
names = [trend['name'] for trend in trends]
# put all the names together with a ' ' separating them
trendsName = "\n".join(names)
print(trendsName)
```

```
Dória
#ال_سعود_ساده_وقاده
#وش_بتسوي_لو_صرت_الملك
#DíaDelMédico
Megyn Kelly
#23Oct
#يامحمد_الهمه_شعبك_معك_للقمه
Parejo
Eli Apple
Lucas Vázquez
Baby One More Time
Rodrigo de la Serna
Gnabry
Juremir Machado
Sandra Day O'Connor
Scorpio
로젠메이든
Guatire
#OBomDesseAnoFoi
#AEKFCB
#TEMPO_KAI
#عموري_رباط_صليبي
#YaşadıkçaTürkçüyüz
#غرد_وكانك_في_التسعينات
#ExpectationsAtMidnight
#BielNoPânico
#AtentadoFakeDoPT
#يكرهك_الشخص_اللي
#TuesdayThoughts
#EverybodyShouldGet
#SomethingWickedIn3Words
#Güneşİle3YılDaha
#MUNJUV
#DHDL
#herkesyoluna
#الغا_قناه_الجزيره
#پرسپولیس
#patisseriesongs
#Commissione
#9DaysForSRKDay
#TusoSierraEnLaW
#افريقيا_يا_اهلي
#MVTAznarCasado
#HappyKrystalDay
#خمسين_هكتار_لصغار_المزارعين
#ViswasamPongal2019
```

#السد_بيرسبوليس
#seoul
#iştekatil
#AdayımızYaşarAydın

In [5]:

```
query="#AcheDin" # Enter user or query
```

In [6]:

```
# We create an extractor object:
extractor = twitter_setup()

# We create a tweet list based on query:

tweets = extractor.search(q=query, count=100)

# We create a tweet list based on user

# tweets = extractor.user_timeline(screen_name=query, count=100)

print("Number of tweets extracted: {}.\n".format(len(tweets)))

# We print the most recent 5 tweets:
print("5 recent tweets:\n")
for tweet in tweets[:5]:
    print(tweet.text)
    print()
```

```
Number of tweets extracted: 100.

5 recent tweets:

RT @KeyaSMamma: देखो कितने #AcheDin आगए है..
5 पैसे की कीमत अब 2800₹ हो गई..😂🙈😝 https://t.co/2DJArwMXLF (https://t.
co/2DJArwMXLF)

RT @sujoysahu: Not for those who have their "ostrich" heads burried in
 the sands of #achedin. India ranks 103 on global hunger https://t.co…
 (https://t.co…)

This is shocking if true ... slowly it's coming up that Gujarat develo
pment model was a facade #Achedin https://t.co/MpZ2tbB1JD (https://t.c
o/MpZ2tbB1JD)

RT @rakeshz: Salman Khan runs over pedestrians on footpath - gets acqu
itted of all the charges.

People trespassing &amp;amp; jaywalking on tracks…

@krazyfrog Bhaiyon aur behno.. #achedin
```

In [7]:

```
# We create a pandas dataframe as follows:
data = pd.DataFrame(data=[tweet.text for tweet in tweets], columns=['Tweets'])

# We display the first 10 elements of the dataframe:
display(data.head(10))
```

	Tweets
0	RT @KeyaSMamma: देखो कितने #AcheDin आगए है..\n...
1	RT @sujoysahu: Not for those who have their "o...
2	This is shocking if true ... slowly it's comin...
3	RT @rakeshz: Salman Khan runs over pedestrians...
4	@krazyfrog Bhaiyon aur behno.. #achedin
5	@AmitShah @BJP4India Finelly i have to see #a...
6	RT @INCKarnataka: #Chowkidar WakeUp! \n\nIs th...
7	@pappu_can_dance @MGNTheTwin How much does Neh...
8	@ndtv A new order and value system? #Achedin?
9	@abhisar_sharma #achedin

In [8]:

```
# Internal methods of a single tweet object:
print(dir(tweets[0]))
```

```
['__class__', '__delattr__', '__dict__', '__dir__', '__doc__', '__eq_
_', '__format__', '__ge__', '__getattribute__', '__getstate__', '__gt_
_', '__hash__', '__init__', '__le__', '__lt__', '__module__', '__ne_
_', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__setattr_
_', '__sizeof__', '__str__', '__subclasshook__', '__weakref__', '_ap
i', '_json', 'author', 'contributors', 'coordinates', 'created_at', 'd
estroy', 'entities', 'extended_entities', 'favorite', 'favorite_coun
t', 'favorited', 'geo', 'id', 'id_str', 'in_reply_to_screen_name', 'in
_reply_to_status_id', 'in_reply_to_status_id_str', 'in_reply_to_user_i
d', 'in_reply_to_user_id_str', 'is_quote_status', 'lang', 'metadata',
'parse', 'parse_list', 'place', 'possibly_sensitive', 'retweet', 'retw
eet_count', 'retweeted', 'retweeted_status', 'retweets', 'source', 'so
urce_url', 'text', 'truncated', 'user']
```

Adding relevant info to our dataframe ¶

In [9]:

```
# We add relevant data:
data['len']  = np.array([len(tweet.text) for tweet in tweets])
data['ID']   = np.array([tweet.id for tweet in tweets])
data['Date'] = np.array([tweet.created_at for tweet in tweets])
data['Source'] = np.array([tweet.source for tweet in tweets])
data['Likes']  = np.array([tweet.favorite_count for tweet in tweets])
data['RTs']    = np.array([tweet.retweet_count for tweet in tweets])
data['Language'] = np.array([tweet.lang for tweet in tweets])
```

In [10]:

```
# Display of first 10 elements from dataframe:
display(data.head(10))
```

	Tweets	len	ID	Date	Source	Likes	RTs	Language
0	RT @KeyaSMamma: देखो कितने #AcheDin आगए है..\n...	103	1054791678781870080	2018-10-23 17:48:27	Twitter for Android	0	12	hi
1	RT @sujoysahu: Not for those who have their "o...	140	1054789273755533313	2018-10-23 17:38:54	Twitter Web Client	0	3	en
2	This is shocking if true ... slowly it's comin...	127	1054787209675468800	2018-10-23 17:30:42	Twitter for Android	2	0	en
3	RT @rakeshz: Salman Khan runs over pedestrians...	144	1054773507874349057	2018-10-23 16:36:15	Twitter for Android	0	2	en
4	@krazyfrog Bhaiyon aur behno.. #achedin	39	1054761799504052225	2018-10-23 15:49:44	Twitter for Android	0	0	hi
5	@AmitShah @BJP4India Finelly i have to see #a...	53	1054760767105290240	2018-10-23 15:45:38	Twitter for Android	0	0	en
6	RT @INCKarnataka: #Chowkidar WakeUp! \n\nIs th...	140	1054738982347685888	2018-10-23 14:19:04	Twitter for iPhone	0	199	en
7	@pappu_can_dance @MGNTheTwin How much does Neh...	77	1054721514606448642	2018-10-23 13:09:39	Twitter Lite	0	0	en
8	@ndtv A new order and value system? #Achedin?	45	1054717987159269378	2018-10-23 12:55:38	Twitter for iPhone	0	0	en
9	@abhisar_sharma #achedin	24	1054689401043804160	2018-10-23 11:02:03	Twitter for iPhone	0	0	und

In [11]:

```
# We extract the mean of lenghts:
mean = np.mean(data['len'])

print("The lenght's average in tweets: {}".format(mean))
```

```
The lenght's average in tweets: 113.69
```

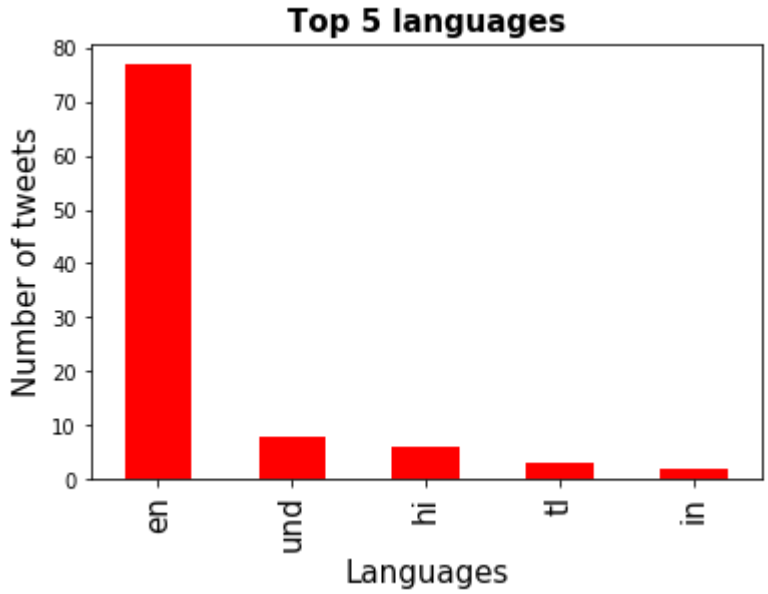
In [12]:

```
tweets_by_lang = data['Language'].value_counts()

fig, ax = plt.subplots()
ax.tick_params(axis='x', labelsize=15)
ax.tick_params(axis='y', labelsize=10)
ax.set_xlabel('Languages', fontsize=15)
ax.set_ylabel('Number of tweets' , fontsize=15)
ax.set_title('Top 5 languages', fontsize=15, fontweight='bold')
tweets_by_lang[:5].plot(ax=ax, kind='bar', color='red')
```

Out[12]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x11477edd8>
```



In [13]:

```
# We extract the tweet with more FAVs and more RTs:

fav_max = np.max(data['Likes'])
rt_max  = np.max(data['RTs'])

fav = data[data.Likes == fav_max].index[0]
rt  = data[data.RTs == rt_max].index[0]

# Max FAVs:
print("The tweet with more likes is: \n{}".format(data['Tweets'][fav]))
print("Number of likes: {}".format(fav_max))
print("{} characters.\n".format(data['len'][fav]))

# Max RTs:
print("The tweet with more retweets is: \n{}".format(data['Tweets'][rt]))
print("Number of retweets: {}".format(rt_max))
print("{} characters.\n".format(data['len'][rt]))
```

```
The tweet with more likes is:
#Bhakths reaction after listening to #MannKiBaat for 4.5 year's 😂😂😂
😂🙏

#achedin #MannKiBaat
#2CroreJobs… https://t.co/VtQkdiGXpb (https://t.co/VtQkdiGXpb)
Number of likes: 27
128 characters.

The tweet with more retweets is:
RT @INCKarnataka: #Chowkidar WakeUp!

Is this the #AcheDin you promised ?
“Nitin Sandesara” hails from Gujarat who is behind ₹5,000 Cr ste…
Number of retweets: 199
140 characters.
```

Time series of Tweets

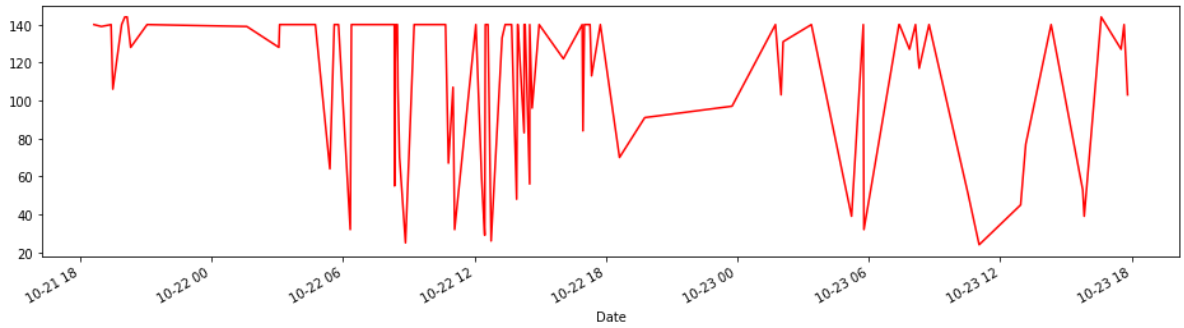
In [14]:

```
# We create time series for data:

tlen = pd.Series(data=data['len'].values, index=data['Date'])
tfav = pd.Series(data=data['Likes'].values, index=data['Date'])
tret = pd.Series(data=data['RTs'].values, index=data['Date'])
```

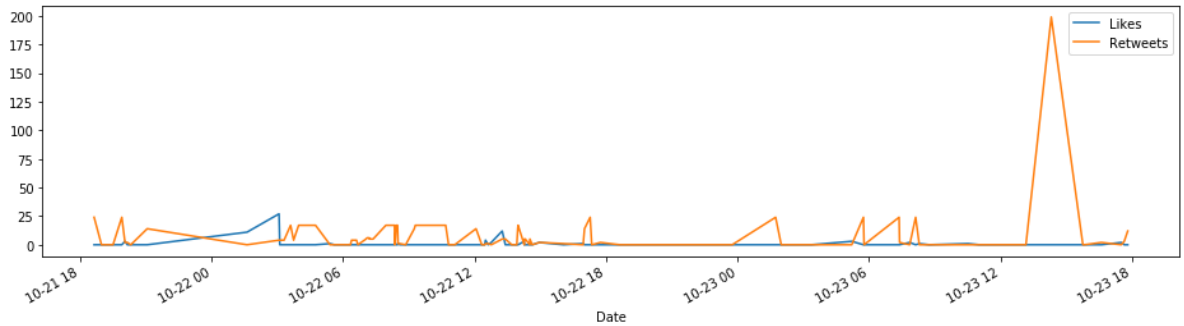
In [15]:

```
# Lenghts along time:
tlen.plot(figsize=(16,4), color='r');
```



In [16]:

```
# Likes vs retweets visualization:
tfav.plot(figsize=(16,4), label="Likes", legend=True)
tret.plot(figsize=(16,4), label="Retweets", legend=True);
```



Pie charts of sources

In [17]:

```
# We obtain all possible sources:
sources = []
for source in data['Source']:
    if source not in sources:
        sources.append(source)

# We print sources list:
print("Creation of content sources:")
for source in sources:
    print("* {}".format(source))
```

```
Creation of content sources:
* Twitter for Android
* Twitter Web Client
* Twitter for iPhone
* Twitter Lite
* Facebook
* Twitter for iPad
* TweetDeck
* WioCities1
```

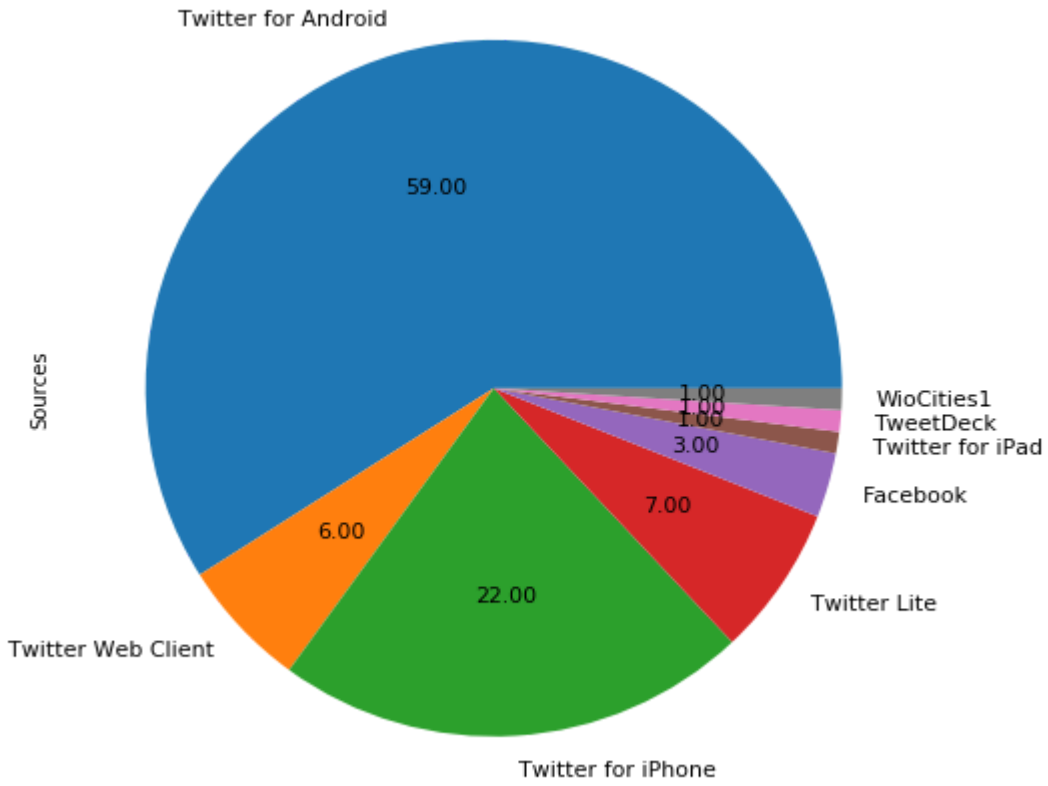
In [18]:

```
# We create a numpy vector mapped to labels:
percent = np.zeros(len(sources))

for source in data['Source']:
    for index in range(len(sources)):
        if source == sources[index]:
            percent[index] += 1
            pass

percent /= 100

# Pie chart:
pie_chart = pd.Series(percent, index=sources, name='Sources')
pie_chart.plot.pie(fontsize=11, autopct='%.2f', figsize=(10,8));
```



Sentiment analysis

In [19]:

```
from textblob import TextBlob
import re

def clean_tweet(tweet):
    '''
    Utility function to clean the text in a tweet by removing
    links and special characters using regex.
    '''
    return ' '.join(re.sub("(@[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)", " ", t

def analize_sentiment(tweet):
    '''
    Utility function to classify the polarity of a tweet
    using textblob.
    '''
    analysis = TextBlob(clean_tweet(tweet))
    if analysis.sentiment.polarity > 0:
        return 1
    elif analysis.sentiment.polarity == 0:
        return 0
    else:
        return -1
```

In [20]:

```
# We create a column with the result of the analysis:
data['SA'] = np.array([ analize_sentiment(tweet) for tweet in data['Tweets'] ])

# We display the updated dataframe with the new column:
display(data.head(10))
```

	Tweets	len	ID	Date	Source	Likes	RTs	Language	SA
0	RT @KeyaSMamma: देखो कितने #AcheDin आगए है..\n...	103	1054791678781870080	2018-10-23 17:48:27	Twitter for Android	0	12	hi	0
1	RT @sujoysahu: Not for those who have their "o...	140	1054789273755533313	2018-10-23 17:38:54	Twitter Web Client	0	3	en	0
2	This is shocking if true ... slowly it's comin...	127	1054787209675468800	2018-10-23 17:30:42	Twitter for Android	2	0	en	-1
3	RT @rakeshz: Salman Khan runs over pedestrians...	144	1054773507874349057	2018-10-23 16:36:15	Twitter for Android	0	2	en	0
4	@krazyfrog Bhaiyon aur behno.. #achedin	39	1054761799504052225	2018-10-23 15:49:44	Twitter for Android	0	0	hi	0
5	@AmitShah @BJP4India Finelly i have to see #a...	53	1054760767105290240	2018-10-23 15:45:38	Twitter for Android	0	0	en	0
6	RT @INCKarnataka: #Chowkidar WakeUp! \n\nIs th...	140	1054738982347685888	2018-10-23 14:19:04	Twitter for iPhone	0	199	en	-1
7	@pappu_can_dance @MGNTheTwin How much does Neh...	77	1054721514606448642	2018-10-23 13:09:39	Twitter Lite	0	0	en	1
8	@ndtv A new order and value system? #Achedin?	45	1054717987159269378	2018-10-23 12:55:38	Twitter for iPhone	0	0	en	1
9	@abhisar_sharma #achedin	24	1054689401043804160	2018-10-23 11:02:03	Twitter for iPhone	0	0	und	0

In [21]:

```
# We construct lists with classified tweets:

pos_tweets = [ tweet for index, tweet in enumerate(data['Tweets']) if data['SA'][inc
neu_tweets = [ tweet for index, tweet in enumerate(data['Tweets']) if data['SA'][inc
neg_tweets = [ tweet for index, tweet in enumerate(data['Tweets']) if data['SA'][inc
```

In [22]:

```
# We print percentages:

print("Percentage of positive tweets: {}%".format(len(pos_tweets)*100/len(data['Twee
print("Percentage of neutral tweets: {}%".format(len(neu_tweets)*100/len(data['Tweet
print("Percentage of negative tweets: {}%".format(len(neg_tweets)*100/len(data['Twee
```

```
Percentage of positive tweets: 35.0%
Percentage of neutral tweets: 60.0%
Percentage of negative tweets: 5.0%
```

In [23]:

```
# Data to plot
labels = 'Positive Tweets', 'Neutral Tweets','Negative Tweets'
sizes = [len(pos_tweets)*100/len(data['Tweets']), len(neu_tweets)*100/len(data['Twee
colors = ['gold', 'yellowgreen', 'lightcoral']
explode = (0.07, 0.07, 0.07)  # explode 1st slice

# Plot
plt.figure(figsize=(10,6))
plt.pie(sizes, explode=explode, labels=labels, colors=colors,
        autopct='%1.1f%%', shadow=True, startangle=140)
plt.axis('equal')
plt.show()
```