Webscraping and sentiment analysis

Problem Statement

The IMDB website (http://m.imdb.com/feature/bornondate) provides a list of celebrities born on the current date.

We first get the list of these celebrities from this webpage using web scraping (the ones that are displayed i.e top 10).

Once we have this list, we run a sentiment analysis on twitter for each celebrity and finally the output should be in the below format

1. Name of the celebrity:
2. Celebrity Image:
3. Profession:
4. Best Work:
5. Overall Sentiment on Twitter: Positive, Negative or Neutral

Solution overview

Main steps :

- Web scraping to get short list of celebrities.
- Use Twitter development account to extract tweets for those celebrities.
- Use sentiment files to estimate sentiment of tweets per celebrity

Main challenges:

- Web scraping a real web page can cause a challenge when it contains jquery scripts that run slowly
- Twitter rate limiting will throw some unexpected errors
- Sentiment analysis is a very complex topic

Web scraping

The URL (http://m.imdb.com/feature/bornondate) is using a delayed jquery script execution. This makes it hard to use the straightforward requests.get method. An option that has been explored is to install the PhantomJS driver and use the selenium package with implicit waiting. This works and the result can be given to BeautifulSoup.

However, investigating the html page in Chrome development tools (network tab) reveals that the page is actually doing a second call to http://m.imdb.com/feature/bornondate_json and that returns directly the required list of celebrities in a nice and easy JSON format. This simple solution has been chosen and the result saved in a pandas dataframe.

Twitter extract

You need to have your own customer and access tokens in order to run the script. Store your secret Twitterkeys into a file TwitterKeys.py in the current working directory.

The tweety package is used to extract tweets using the full name of the celeb as simple search query. Tweets are typically quite messy and trying more sophisticated queries will not necessarily improve the analysis.

There is a constant NUMBEROFROWS in the script that defines the maximum number of tweets per celeb.

Sentiment analysis

A real sentiment analysis is a complex matter and would require some advanced supervised learning methods. This project takes a shortcut and relies on the NLTK package.

The magical trick is in

```
from nltk.sentiment.vader import SentimentIntensityAnalyzer
```

Check out the example in http://www.nltk.org/howto/sentiment.html to see how it works.

Executing the script

Go to github https://github.com/stefMT2970/WebScrapingSentiment

Download WebScrapingSentiment.py to your local development environment and examine the code.

References

1. NLTK : Bird, Steven, Edward Loper and Ewan Klein (2009), Natural Language Processing with Python. O'Reilly Media Inc.