**CS410 Course Project – Movie Review Analysis**

**Installation**

1. Download the python scripts from github
2. Install Python Packages. Use Command line to install packages using pip.

# Ensure your pip is up to date

pip install --upgrade pip

# Install Beautiful Soup

pip install beautifulsoup4

For issues – Refer <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

# Install Vader Sentiment

pip install vaderSentiment

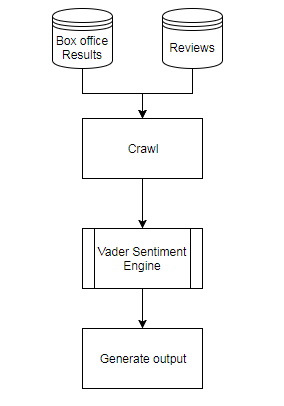
For issues – Refer <https://github.com/cjhutto/vaderSentiment>

# Install NT

pip install -U nltk

For issues – Refer <http://www.nltk.org/install.html>

**Flow Chart**

****

**Architecture**

1. **Getting Box office Results for the weekend**

Script Name – crawlbasic.py

Input – None

Output - movienumbers.txt

[www.the-numbers.com](http://www.the-numbers.com) is a website that provides the weekend box office results.

The crawlbasic.py script crawls the top 15 movies of the weekend from the following url “http://www.the-numbers.com/weekend-box-office-chart” using Beautiful Soup package and creates a output text file movienumbers.txt. Following are the fields in the output file for each of the top 15 movies

1. Movie Name as in the numbers website
2. Gross Revenue for the current weekend
3. Revenue Change from last week in percent
4. Total Gross Revenue for the movie
5. No. of week since movie release
6. Tag links of the numbers website for future reference

Timing – This script can be executed independently. However, the website is updated with the weekend box office information normally on Mondays or early Tuesdays. So the script needs to be executed post the update of the results.

1. **Downloading the reviews for Top 15 Movies**

Script Name – crawlrt.py

Input – None

Output - moviertlist.txt, 30 movie critics and user review csv files for top 15 movies

www.rottentomatoes.com is a website that captures reviews from critics and users.

The crawlrt.py script crawls the critics and user reviews for the top 15 movies of the weekend from the following url “https://www.rottentomatoes.com/browse/in-theaters/” using Beautiful Soup package. The scripts creates individual pipe separated text files containing the reviews from the critics and users. The script also creates a list of movies for which the reviews have been downloaded.

Timing – This script can be executed independently. However, the rotten tomatoes website is updated with the current weekend movies normally on Mondays. So the script needs to be executed post the update of the results.

1. **Creating an Input file for Sentiment Analysis**

Script Name – mergefiles.py

Input – movienumbers.txt, moviertlist.txt

Output - movieinput.txt

In order to review the timing dependency of both the websites, a merge process to create an input file to create the top 15 movies from both the websites.

The mergefiles.py script combines the top 15 movies from both the websites into one text file movieinput.txt. The review of this input file helps validating any timing dependencies of both the crawlbasic.py and crawlrt.py scripts.

Timing – This script needs to be executed after crawlbasic.py and crawlrt.py scripts.

1. **Running Sentiment Analysis**

Script Name - Sentiment Analysis.py

Input - movieinput.txt, 30 Movie reviews (critics and user) from top 15 movies

Output - movieoutput.txt, 30 movie critics and user review scores

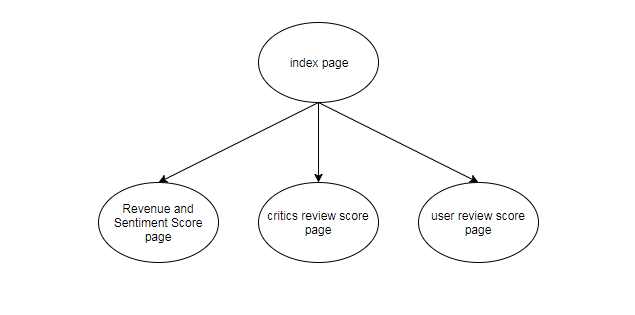
The Sentiment Analysis.py script takes movieinput.txt file and the review files as input and generates the score using vader sentiment package for the individual reviews. The script also generates consolidated results for each movie and creates an output file. The output file movieoutput.txt contains the following fields

1. Movie Name in rotten tomatoes
2. Movie Name in the numbers site
3. Gross Revenue for the current weekend
4. Revenue Change from last week in percent
5. Total Gross Revenue for the movie
6. No. of week since movie release
7. Tag links of the numbers website for future reference
8. Avg. Sentiment Score for critics review
9. Avg. Sentiment Score for user review
10. Min. Score of critics reviews
11. Max. Score of critics reviews
12. Min. Score of user reviews
13. Max. Score of user reviews
14. No. of positive score for critics reviews
15. No. of negative score for critics reviews
16. No. of positive score for user reviews
17. No. of negative score for user reviews
18. **Generating UI based on Output File**

Script Name – htmlgenerator.py

Input – movieoutput.txt, 30 movie critics and user review scores

Output – index.html, Critics Review Score html Pages, User Review Score html Pages, Revenue projection and consolidated Score html page



The htmlgenerator.py script would generate the following html files

* 1. Index Page – The main html that list the top 15 movies for the week and list the average sentiment score.
  2. Critics Review Score Pages – Html page for each movie listing the critics review and the sentiment score for each of the review.
  3. User Review Score Pages – Html page for each movie listing the user review and the sentiment score for each of the review.
  4. Revenue projection and consolidated Score page – Html page showing the revenue graph from the-numbers website for the movie and the showing the consolidated values for each movie

**Use Case - A Simplified Version without Crawling**

1. Download the python scripts and sample.zip
2. Unzip sample.zip to download the sample movieinput.txt file and the review csv files
3. Install the required python packages
4. Run Sentiment Analysis.py.
5. Run htmlgenerator.py
6. Open index.html

**Use Case – Detailed Version**

1. Download the python scripts
2. Install the required python packages
3. Run crawlbasic.py
4. Run crawlrt.py
5. Run mergefiles.py
6. Run Sentiment Analysis.py.
7. Run htmlgenerator.py
8. Open index.html