**World Quant University**

**Professor: Tiberiu Stoica**

**Machine Learning**

Nikolas Lippmann Pareschi - [nikolaslippmann@gmail.com](mailto:nikolaslippmann@gmail.com)

**Final Project**

### Overview

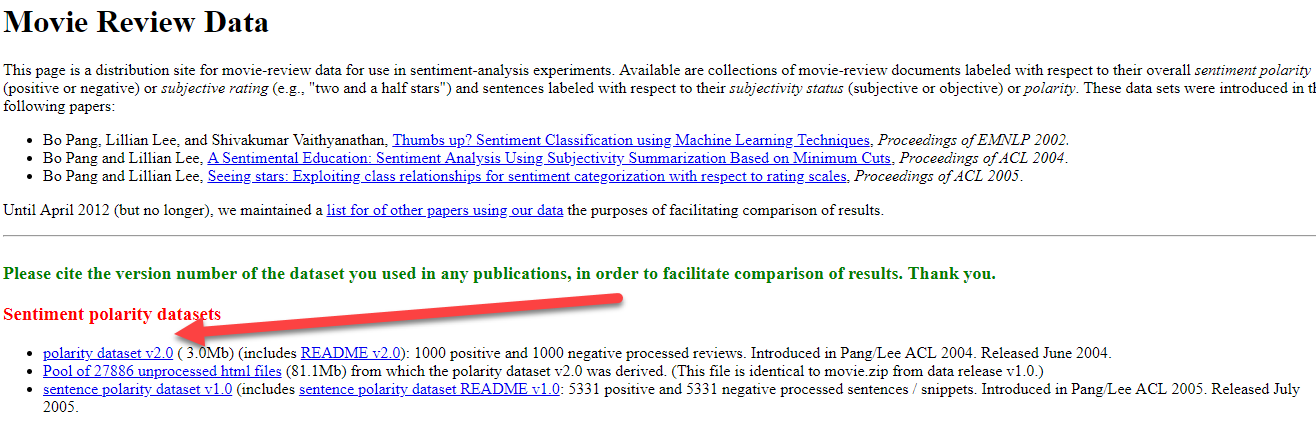
In the lab Text Mining and Sentiment Analysis, we used the Naïve bayes classifier to see ratios of negative and positive sentiment.  In this project, we will be analyzing the same movie review corpus, but this time we will employ a frequency-inverse document frequency(tf-idf) for analyzing the word cloud.

The tf-idf metric can give us ranking weights for words in our corpus. Its value is proportional to the number of occurrences of a word(corresponds to term frequency) in a particular document. However, it is also inversely proportional to the number of documents in the corpus(corresponds to the inverse document frequency), where the word occurs. The tf-idf value is the product of term frequency and inverse document frequency.

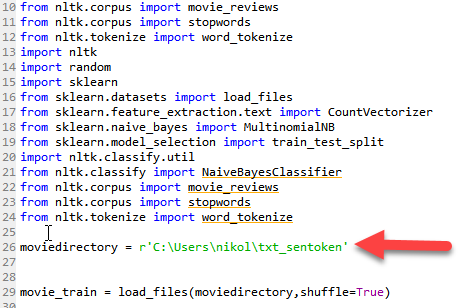
**Exercise**:  calculate the tf-idf for the movie\_review corpus and the top ranking words for that corpus.

When you run the code from the file, make sure you follow the readme instructions. In the readme file the user is guided regarding where to put the movie files. Please download the files from:

<https://www.cs.cornell.edu/people/pabo/movie-review-data/>

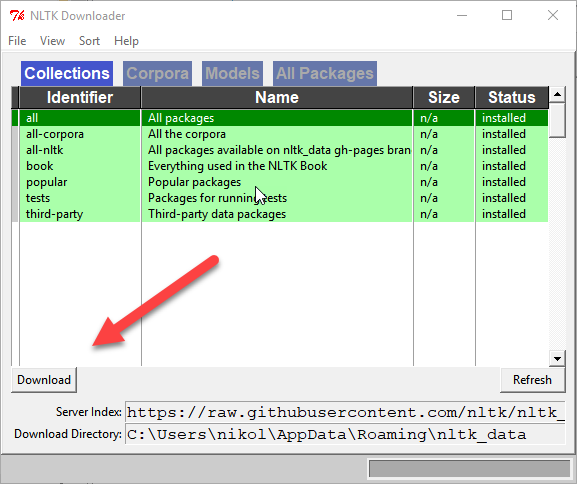


And place them in your working directory:

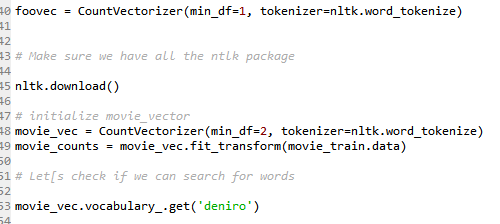


Also unpack the positive files and negative files to the txt\_sentoken folder. This is mandaroty.

When you run the code the NTLK Downloader will open. Please hit the download button and after downloading close the window:



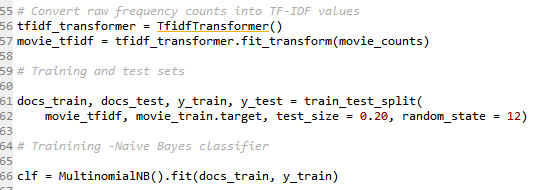
We initiatilize the movie vector and check if we can find any word:



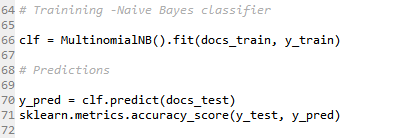
Result:



We then convert the frequencies to the TF-IDF algorithm and then apply it the Nayve Bayes in the training set:



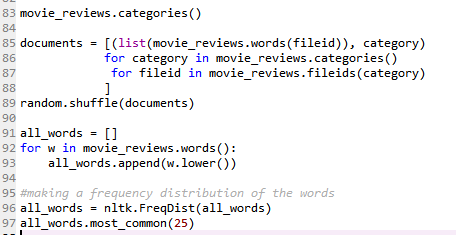
We now make some predictions:



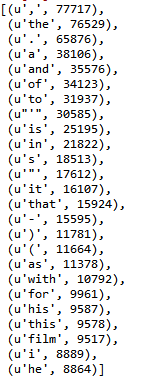
Accuracy:



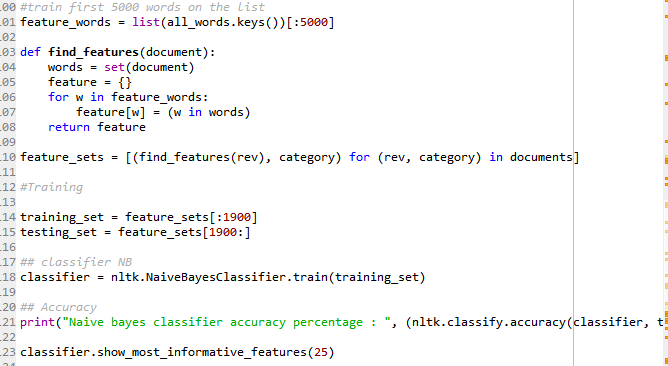
We look into the most common words:



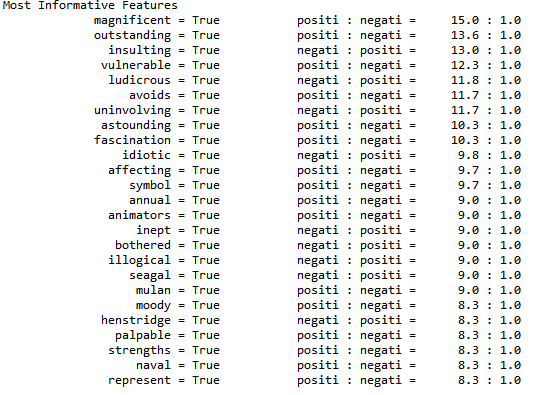
Results:



We use the NB to look into the most informative words:



Results:



Regarding how to improve the algorithm, I have searched this site in which the data scientist uses Support Vector Machine instead of Bayes:

<https://appliedmachinelearning.blog/2017/02/12/sentiment-analysis-using-tf-idf-weighting-pythonscikit-learn/>

His accuracy using SVM was superior than using Bayes.