

# Movie Reviews Sentiment Classification

Dongyu Zhao (dongyuzhao@brandeis.edu)

## Overview

Movie reviews are very common nowadays. For sentimental analysis, there're two categories of the reviews: positive and negative. In this final project, I'll follow the guide of the paper *Thumbs up? Sentiment Classification using Machine Learning Techniques*<sup>1</sup> to train a model to classify a movie review of plain text into positive or negative using Machine learning algorithm (Naïve Bayes or SVM) and movie reviews corpus in NLTK.

## Dataset

- movie\_reviews: A corpus in NLTK of 2k movie reviews with sentiment polarity classification.
- stop\_words: A corpus in NLTK of 2,400 stopwords for 11 languages.
- negation\_words: Negation words such as “not”, “no”, “never”. Follows the idea of Das and Chen (2001)<sup>2</sup>, add a “NEG-” label before words from a word in negation\_words to the end of that sentence. Therefore, any word might have two formats, the original word or “NEG-” + word. For example, after adding the “NEG-” label, words of sentence [“Standard”, “English”, “does”, “not”, “have”, “two”, “negatives”, “in”, “the”, “same”, “clause”] shifts to [“Standard”, “English”, “does”, “not”, “NEG-have”, “NEG-two”, “NEG-negatives”, “NEG-in”, “NEG-the”, “NEG-same”, “NEG-clause”]. However, the “NEG-” tag doesn't have to mean that the word changes its sentiment polarity (from negative to positive or opposite). It just give the word another format such that if there's difference of this word's two format in sentiment analysis, it can be independently analyzed.

## Preprocess Data

- Loop through each token in the movie\_reviews corpus and wipe the stop words of each reviews if the word in stopwords.words(“english”) for further analysis
- Loop through each sentence in the movie\_reviews corpus and then loop through each word in each sentence, if a word in negation\_words appear, then add the string “NEG-” before each word from the negation word (not included) to the end of the sentence.

---

<sup>1</sup> <http://www.aclweb.org/anthology/W02-1011>

<sup>2</sup> Sanjiv Das and Mike Chen. 2001. Yahoo! For Amazon: Extracting market sentiment from stock message boards. In *Proc. of the 8th Asia Pacific Finance Association Annual Conference (APFA 2001)*.

## Features

I use word presence to build the features from words. That is for the features dictionary, make a feature equal to 1 if that word format appears in the review, otherwise, set the value to 0. E.g. if "features" appears while "NEG-features" not in a review of one preprocessed review, then  $\text{features}[\text{"features"}] = 1$  and  $\text{features}[\text{"NEG-features"}] = 0$ .

- Unigram: Tokenize words of the total corpus. To avoid misspelled words or meaningless number combination such as "looooot" or "35s". I only consider tokens which appears more than 5 times.
- Bigrams: Adjacent tokens pair in each review of the total corpus. For the same reason mentioned above, I only consider tokens pair which appears more than 3 times. The feature name of a pair ("word1", "word2") is "word1+word2" e.g. "n't+NEG-a", so it shares a one-to-one relationship to the pair so there's no repetition.
- Part of Speech: I use the `nlk.pos_tag()` function to append POS tags to each word. Of course this is done before append "NEG-" tags, so for each word, the two formats of the word shares the same POS tag if it in similar context. Feature name of the POS pair ("word", "POS") is "word+POS" e.g. "NEG-features-NNS". Also, I only consider POS pairs which appear more than 5 times

In addition, intuitionally, the adj words (words with POS tag "JJ") are more important in the role to classify, so I create a feature dictionary only consists adj words. To compare this, I also create a feature dictionary which consists the most frequent words (except for punctuation in `string.punctuation`) with same number feature of the former one.

## Models

- Naïve Bayes

Classify the review to category  $c = \text{argmax}_c P(c|d)$ , where

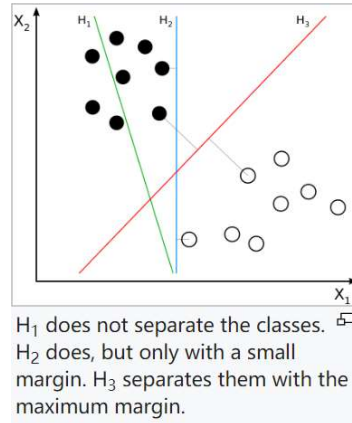
$$P_{\text{NB}}(c|d) := \frac{P(c) \prod_{i=1}^m P(f_i|c)}{P(d)}$$

However, when combining the Unigram and Bigrams feature dictionary together violate the independent assumption of each feature. Therefore, I try another model next.

I use the `nlk.NaiveBayesClassifier` model in my project.

- Support Vector Machines

The main idea of SVM is to find a hyperplane  $\vec{w}$  that separate the review vectors to two categories with maximum margin



I use the Linear SVM model of `sklearn.svm.LinearSVC` in my project.

## Results Analysis











- Split dataset:  
I split the total corpus into `train_set` and `test_set`, the size of the `train_set` is 1000 (half of the total data).
- Cross-validation  
I use the 3-fold cross validation to calculate the average accuracy of each model on different feature sets for the `train_set`. Results list below.

	Features	# of features	NB		SVM	
			train_set	test_set	train_set	test_set
(1)	unigram	14,922	78.1	76.6	<b>81.0</b>	80.6
(2)	bigrams	16,382	<b>68.9</b>	73.4	68.7	72.3
(3)	unigram + bigrams	31,304	79.3	78.4	<b>82.2</b>	81.1
(4)	unigram + POS	15,992	78.2	76.8	<b>80.3</b>	80.8
(5)	adjectives	2,833	<b>75.7</b>	75.8	75.3	75.9
(6)	top 2,833 unigram	2,833	<b>79.5</b>	76.5	79.3	77.3

- Results of `train_set` and `test_set` is quite consistent except for bigrams features, so the overfitting problem is not very evident.
  - Bigrams itself doesn't perform very good compared to the unigram features. But combining both unigram and bigrams features do perform better than the unigram itself, and this becomes the most powerful model in these 6 cases.
  - Though intuitively, the adjectives play more important role in the sentiment polarity, however, the top 2, 833 unigram model performs better. More interesting, in the Naïve Bayes model, the top 2,833 unigram model overcomes the total unigram (only consider appear more than 5 times), but in the SVM model, the total unigram model performs better.
- Prediction  
I download some reviews from Rotten Tomatoes<sup>3</sup>, each review has a rating. I choose the 5 positive reviews and 5 negative reviews from the website and use the SVM models for

<sup>3</sup> [https://www.rottentomatoes.com/critics/latest\\_reviews/](https://www.rottentomatoes.com/critics/latest_reviews/)

using unigram + bigrams feature sets to predict, Since I use the 3-fold cross validation, so each model with one feature set has 3 classifiers. Each time, I let the computer randomly choose one classifier. Results are list below.

	Overview	Prediction
positive-1	 7/10 Islam and the Future of Tolerance "Anyone can wrangle a group of talking heads in front of a camera or audience to blaviate on the merits of religious belief, but Avila and Shapiro's purposes are more urgent and pressing." <small>Posted Dec 18, 2018 9:12 AM UTC</small> <b>Nathanael Hood</b> The Young Folks	positive correct
positive-2	 4/5 Okja "Pigs are clever and if they were pretty too we probably wouldn't eat them. Guilt and innocence marble this like fat in a prime cut, with superpig Okja caught in the middle. Funny and sad, it's an allegory about as subtle as, well, a giant farting pig." <small>Posted Dec 18, 2018 9:12 AM UTC</small> <b>Sarah Cartland</b> Caution Spoilers	positive correct
positive-3	 Vice "It's a character study of a man imbued with a guiding vision of his own greatness that, finally, yields a somewhat scattered, just-OK movie." <small>Posted Dec 18, 2018 8:12 AM UTC</small> <b>Brian Lowry</b> CNN.com	positive correct
positive-4	 5/5 Mary Poppins Returns "It's a thing of beauty, intelligence, and social consciousness that evokes not just a sense of wonder, but also of infinite the possibilities that changing a point of view can provide." <small>Posted Dec 19, 2018 7:12 AM UTC</small> <b>Andrea Chase</b> Killer Movie Reviews	positive correct
positive-5	 10/10 Spider-Man: Into the Spider-Verse "The story itself is fantastic, full of humor and heart as Miles not only has to cope with events far bigger than himself with more experienced heroes, but learns to come into his own and embrace life as a superhero himself." <small>Posted Dec 19, 2018 7:12 AM UTC</small> <b>Andrea Thompson</b> The Young Folks	positive correct
negative-1	 1/4 Welcome to Marwen "'Welcome to Marwen' does not work as a drama of addiction, and frankly it doesn't work as a celebration of Hogancamp's creations, which work best as stunning still-photo images." <small>Posted Dec 19, 2018 12:12 PM UTC</small> <b>G. Allen Johnson</b> San Francisco Chronicle	negative correct
negative-2	 4/10 Mary Poppins Returns "It's a waste of a practically perfect character, even in its winning moments." <small>Posted Dec 18, 2018 6:12 AM UTC</small> <b>Sean Collier</b> Pittsburgh Magazine	negative wrong
negative-3	 C- Ben is Back "Hedge's script never seems willing to fully immerse itself into the darkness. It is content with simply dipping its toes in at various points." <small>Posted Dec 19, 2018 7:12 AM UTC</small> <b>Courtney Small</b> Cinema Axis	positive wrong
negative-4	 1/4 First Reformed "... making viewers feel as numb inside as the main character... feels like its missing a necessary point-of-view, leaving all else as an exercise in futility... plus a somewhat-cool one-sheet poster." <small>Posted Dec 19, 2018 6:12 AM UTC</small> <b>Kevin A. Ranson</b> MovieCrypt.com	negative correct
negative-5	 C- Aquaman "While Momoa is just fine as the titular 'Aquaman,' [the] screenplay features far too little humor and far too much origin story - by the time Orm's visiting those remaining undersea realms, they feel like jammed in afterthoughts." <small>Posted Dec 19, 2018 5:12 AM UTC</small> <b>Laura Clifford</b> Reeling Reviews	negative correct

Note: The data for prediction is not the overview, but the whole review which can be accessed by clicking the links

## Further Work

- Presence or frequency or tf-idf?  
I only generate feature for presence, but there're lots of other option such as the frequency or the tf-idf value of the word.
- Useless features problem  
In this project, I don't think too much about choosing the features, some of them might play a more important role while others might be totally useless in classification, the evidence is the result that in Naïve Bayes model, the top 2,833 unigram model outperforms the total unigram (only consider appear more than 5 times). Getting rid of those useless or even misleading features or adding more subtle features might pushes the results better.
- Position of the word also gives some information for the sentiment polarity. For example, most of the reviews will describe some plots in the beginning, words in this part then is not very relevant for sentiment classification. But the problems are 1) The structure is complicated, some reviews like to put their opinions at the beginning, then describe the plot, or some sentences to express their feelings, but others might prefer start by plot directly. 2) How to divide the whole review into different parts. Some may write more about the plot, some may write more about his own opinion. These two problems make it hard to set each part's length as well as its order.