**Naive Bayes**

Introduction

Sentiment analysis is the interpretation and classification of emotions (positive, negative and neutral) within text data using text analysis techniques. Sentiment analysis allows businesses to identify customer sentiment toward products, brands or services in online conversations and feedback. Sentiment analysis is used in various places, like categorizing reviews into good or bad on online shopping websites or categorizing movie reviews, etc.

The problem is to classify the sentences according to their sentiments. According to the data set used, 1 is used to represent positive sentiment and 0 represents negative sentiment. The data set is a .txt file.

Problem Definition and Algorithm

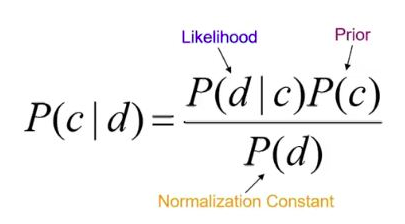
Problem : Using the data to train the model and classify the sentences from the test set as positive or negative sentiment.

Input:

* A document d - a1\_d3.txt
* A fixed set of classes C = {c1, c2} which in this case is {0,1}

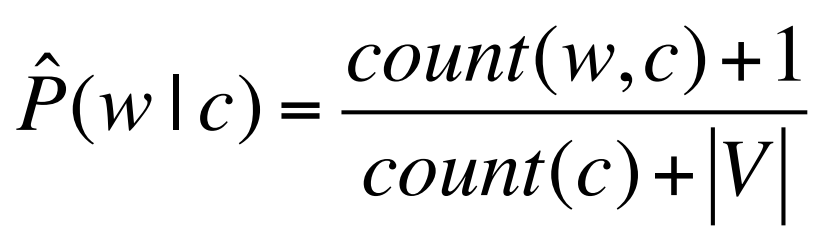
Output: A predicted class c ∈ C

Using Naive Bayes Classification for sentiment analysis:

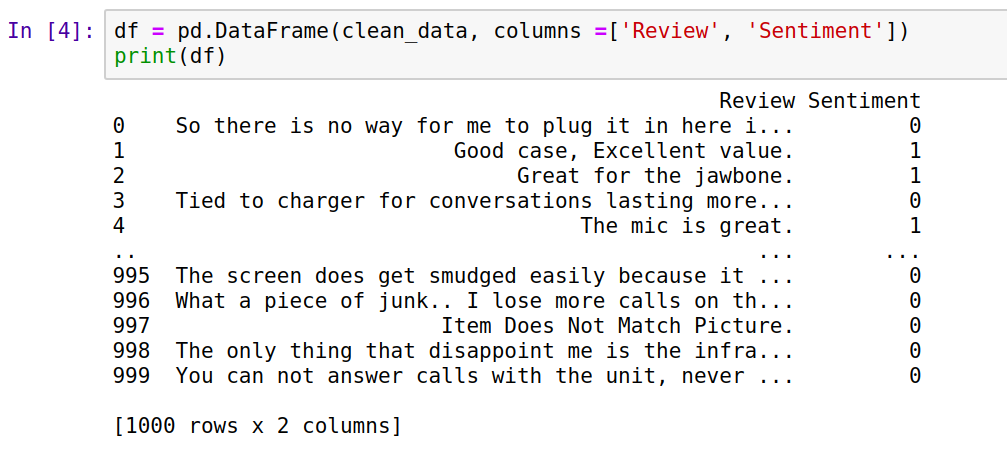


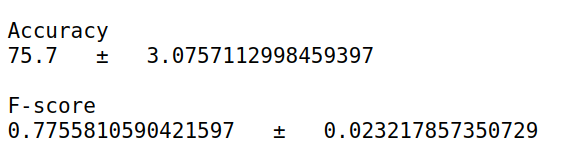
Calculating the likelihood probability using: 

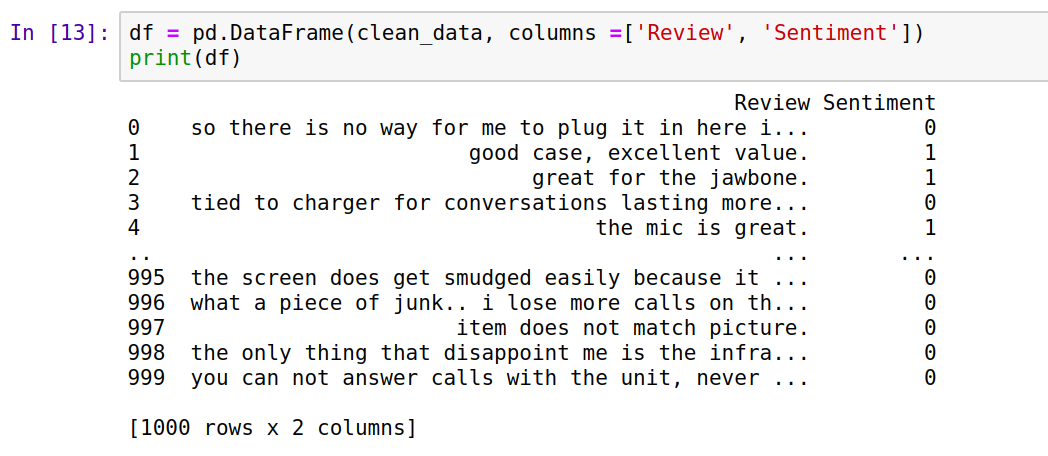
We made predictions according to Naive Bayes algorithm and used Laplace Smoothing to smooth categorical data. Laplace Smoothing is introduced to solve the problem of zero probability. This handled out-of-vocabulary words. By applying this method, prior probability and conditional probability was calculated as: where V is the length of the vocabulary.

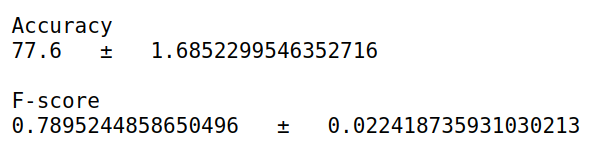


Design Decisions and Result:

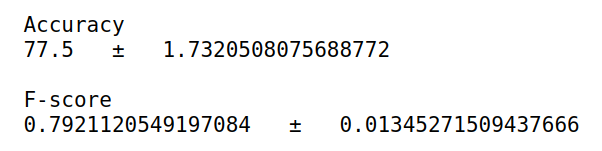
**Case with no text Processing :**

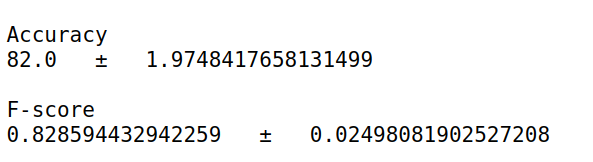
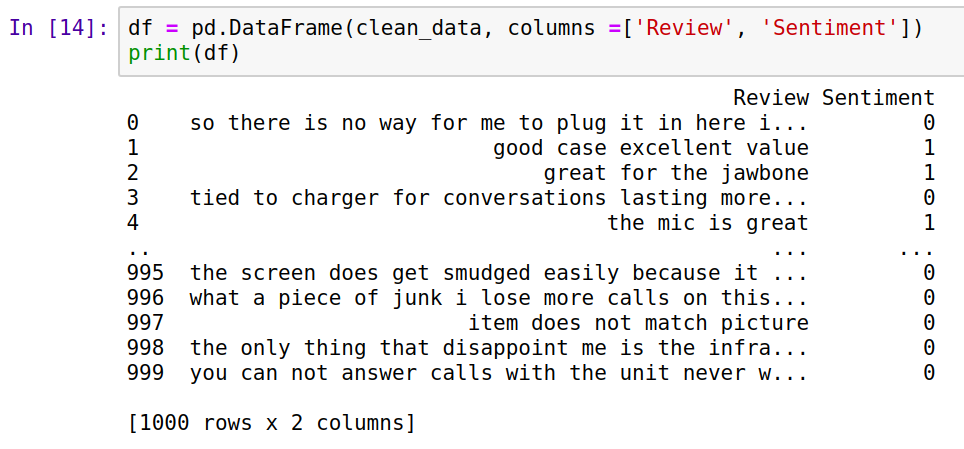


**Model using Text processing as converting the sentences to lower case but does not remove punctuation and symbols:**



**Model using Text processing as removing punctuations but the sentences are not converted to lowercase:**



**Model using Text Processing: Punctuations / symbols are removed and sentences are converted to lowercase**

It can be seen that the model performed better with text processing, and this was the final accuracy and F-score of the model.

Algorithm:

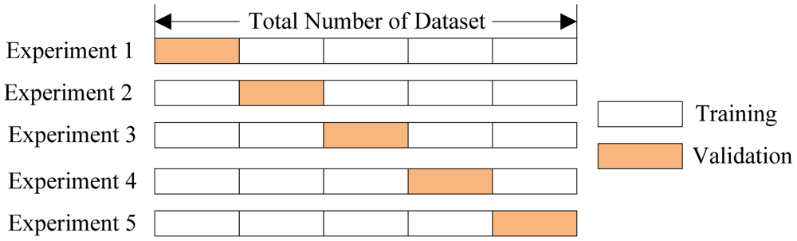
Break the text into sentences

Text Processing

Load the document

Remove punctuations

Convert all the sentences to lower case



5-fold Cross validation

Accuracy

F-score

Evaluation Metrics

Predict

Words and its frequency associated with negative sentiment

Words and its frequency associated with positive sentiment

Store the frequency of each word in the vocabulary

Calculate probabilities

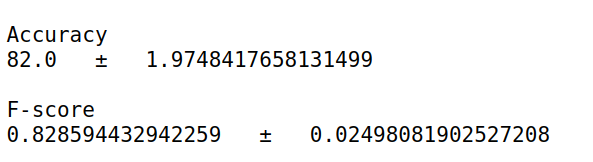
For each review, extract the words and its frequency

Breaking all sentences to words - Define Vocabulary

Test Set

Train Set

Results:

The model performed well. Final evaluation metrics -