Report abou the
CREATION OF A TEXTMINING MODEL
and its implementation to classify film critics

GENÍS LÁINEZ MORENO,

Univertitat Autònoma de Barcelona. 25/03/2020

Abstract

Texmining is a powerful tool that can be easily impelented trough code and it can be very useful to create easy models in order to classify texts into different tematics. In This report is intended to explain the creation of a textmining model done in order to classify film critics.

1 The Dataset

The Dataset we are going to exploit to train the model as well as test consists on 2000 text files where each of them have a critic about a film. Half of them are labeled as 'neg' meaning they are negative critics and half of them labeled as 'pos' meaning they are positive. This labels can be extracted trough the name of the folder that contain each text.

This Dataset has been imported with the help of some python libraries like *glob* and *pandas* into a pandas Dataframe with its label attached. They all the txt have been randomly mixed and then divided into two smaller Datasets, the train Dataset and the test Dataset.

2 Text Preprossessor

Since we are going to exploit the text, It is mandatory to preprocess the data in order to have a consistent and an arbitrary good new Dataset to feed the model. In order to do this prepossessing of data we have used the following concepts:

- **Lemmatization** In order to transform all the words in the way we could find it on a dictionary
- **Stemming** In order to reduce every word into its stem if it has one.
- **Stopwords** In order to delete all the stopwords that do not contribute in our predictions.
- **Categorizing and Tagging words** I have assigned every word its categoy in order to just use the ones that are appropiate. I have used Nouns, adjectives, adverbials, numbers and superlatives.
- **N-grams** I have taken the words in groups of two. The motivation was that, sinse I have take the nouns and adjectives should be more accurate if I pick the words in groups of two to know which noun the adkective referes.

3 Train

The model has been trained on the 90% of the texts that we have. The train gave a weight to some words that are repeated more than 3 times (a parameter that I have been changing to know the best adjustment).

I have created a CountVectorize on the texts labeled as 'pos' and one on the texts labeled as 'neg'. By doing so, I have obtained two bags of words with them weights.

4 Results

Test

I would expect a text to be 'pos' if the sum of its words wighted by 'pos' CountVectorize is larger that the sum done with the 'neg' CounterVectorize. By suposing that I am now ready to to this experiment on the test and obtain some results.

The test has been performed on all the rest of the texts that have not been used for training so the remaining 10% of the total. I have obtained a model of an accuracy of 77% of this dataset that I supose to be representative of the group of all possible critics. A more accurate representation of the results can be obtained by observing the (normalized) confusion matrix on the figure 1

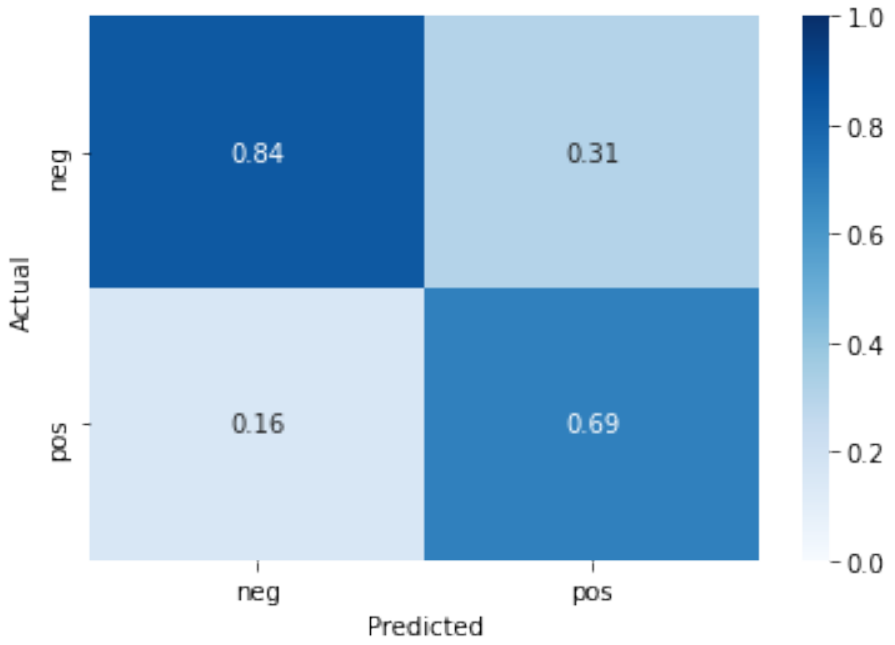

Figure 1: confusion matrix