Readme File- Group 7

Sachin Mohan Sujir, Himanshu Nirmal

An Online Web-Based Application for News Classification Based on Category Using Unique Bag of Words.

\*Created using Python v3.6\*

1. def pre\_processedstr(string):

string: The headline will be processed. All characters except letters will be removed, multiple spaces will be converted to single spaces and the string will be converted to lower case. The headline will be tokenized and stemmed and the stop words will be removed.

2. class NaiveBayes:

NaiveBayes class contains four main functions.

2.1

def bagOfWords(self,headline,dict\_index): This function splits the example on the basis of space, tokenizes it and then adds them to its corresponding bag of word.

headline: an headline from data

dict\_index: implies to which category of bag of word does the headline belong to

def train(self,dataset,labels): This function trains the Naive Bayes model and computes a bag of words for every class.

dataset: training data

labels: category for the trained data

def getHeadlineProb(self,test\_headline): This function calculates the posterior probability for the given headline

test\_headline: the example on which the posterior probability is to be calculated

def train(self,test\_set): This function determines probability of each test headline against all classes and predicts the label against which the class probability is maximum

test\_set: test dataset

**- To run the code, you need to have python v3.6 or higher and have the dataset loaded onto the machine along with the .py file and importantly all the necessary libraries imported.**

The following libraries need to be installed in the machine to run the program-

* Pandas
* Numpy
* Collections, default dicts
* NTLK
* sklearn

- The algorithm is basically built using python version 3.6

- The data\_set used here is news.csv which contains nearly 6000 data with news headline and its corresponding category.

- The x\_train and y\_train takes the headline and the category of the dataset respectively

- The pclasses variable stores the values of classes predicted. For test case useablity, we have our own test cases written to check the predictions. We have designed five test cases to predict the category.

- For real-time test cases, we split the train data into test data and train data and then run the algorithm for the test data.

- The real-time test data is commented off to illustrate how the algorithm works. So, we have defined our own test cases(test data).

ScreenShot of Test Cases

