INFO284 - First Obligatory Assignment
Naives Bayes for classification

than.soe@uib.no

5 February 2020 - Spring 2020

1 Overview

The goal of this group project assignment is to obtain practical knowledge of the Naive Bayes algorithm by creating a Naive Bayes classier from scratch that can predict the sentiment of tweets into three categories, namely: positive, negative or neutral. This type of social media sentiment analysis helps companies keep track of what people are saying about them.

The project can be done in a a group of two or individually. The deadline for the project is *2nd March 2020 at 11:59 a.m.*

2 Description of the data

Source of the data is from Figure Eight (https://www.figure-eight.com/data-for-everyone/) Airline Twitter Sentiment. Download it from here `https://github.com/thanhtut/info284_lab/tree/master/assignment1/twitter-airline-sentiment`. It has 16000 data rows. The description of the data from the provider is as follows.

> A sentiment analysis job about the problems of each major U.S. airline. Twitter data was scraped from February of 2015 and contributors were asked to first classify positive, negative, and neutral tweets, followed by categorizing negative reasons (such as "late flight" or "rude service").

3 Assignment

The assinment cosnsists of the following tasks

1. Clean up the dataset and split into training and test sets.
2. Build a vocabulary of words from all Tweets to be used in representation.

3. Decide whether you can make use of the metadata fields present in the dataset and preprocess the metadata if necessary.

4. The rst is the prior probabilities of the class $P(y = y_k)$. For each class $y_k neutral, negative, positive$, this results to the number of tweets labeled with the particular class from the total number of tweets in the training set.

5. The second set of parameters to estimate are the likelihood probabilities $P(w = w_i|y = y_k)$, for each word w_i in the the vocabulary V (which consists of a set of unique words from all tweets in the training set). The likelihood probability of the word w_i given class y_k is the number of times w_i occurs in a news article with label y_k from the total number of news articles in the training set labeled with y_k .

6. Build your Naive Bias Classifier from scratch (i.e. without using any machine learning libraries that provides a pre-built classifier. Feel free to use any lower level libraries for working with CSV, preprocessing text and plotting)

7. Evaluate your classifier on the test set and calculate the error rate.

8. When a news article contains a word that is not in the dictionary, the posterior class probabilities are zero. To overcome the problem you can use a smoothing technique known as Laplace smoothing and calculate the likelihood probabilities as:

$$\hat{P}(w = w_i|y = y_k) = \frac{\#R\{w = w_i \wedge y = y_k\} + 1}{\#R\{y = y_k\} + |V|}$$

   where the R operator denotes the number of elements in the training set of news articles R that satisfy the constraint in the brackets, and $|V|$ is the cardinality of the vocabulary.

9. Create a command line utility for your classifier that can take a arbitrary tweet and compute the sentiment.

10. Create an explanation generator why a tweet is classified in the specific category.

11. Pick two correct and incorrectly predicted tweets from your test set and explain why your classifier assign a certain label.

    The submission should contain all steps above documented as well as working code. Follow the style guide PEP8 mentioned here `https://www.python.org/dev/peps/pep-0008/`.