CAP6307 Advanced Text Mining **Fall 2020**

Homework 1

Lecturer: Dr. Fei Liu *Due: Tuesday 9/15 11:59PM EST*

Note: *Homework modified from Lisbon machine learning summer school.*

1.1 Text Classification and Naive Bayes (15 points)

The goal of this assignment is for you to gain familiarity with the **multinomial Naive Bayes classifier**. Specifically, you will look into an existing Python-based implementation and fill out the missing code block to gain an understanding of applying multinomial Naive Bayes to text classification.

In the homework package (`HW1.tar.gz`), you are provided with the starter code and a dataset. The code was written in `Python 2.7` and `numpy`. If you would like to install this programming environment, please go to `https://www.continuum.io/downloads`. Download the Anaconda version (with Python 2.7) that is compatible with your operating system. The installation instructions are available at `http://docs.continuum.io/anaconda/install`. Installing Anaconda will install both Python 2.7 and numpy.

There are two data files in the package: `positive.review` and `negative.review`. They correspond to positive and negative book reviews. The text has been preprocessed so that each line contains a review document; each token (e.g., `year:2`) represents a word and its frequency in the document. The last token (e.g., `#label#:negative`) in each line indicates the polarity (label) of the document.

The starter code includes four files: `linear_classifier.py`, `multinomial_naive_bayes.py`, `run_classifier.py`, `sentiment_reader.py`. The functionality of the files should be self-evident..

- `(10 points)` The file `multinomial_naive_bayes.py` currently has a missing code block. Search `TODO` in the file and you will find the missing block. Your task is to fill out the missing code. Upon successful completion of the code, you will run `python run_classifier.py` and this will return the following results: `Accuracy on training set: 0.985625, on test set: 0.687500`.
- `(5 points)` The starter code randomly chooses about 80% the documents to form the training set and the rest as test set. Modify the code so that the train/test split is 50%/50%. After that, run `python run_classifier.py` and report the returned results.

Please submit: A report named `report_firstname_lastname.pdf`. Copy and paste to the report: 1) the missing code block you filled in, and 2) the returned result after modifying the train/test splits.