Programming Assignment 1

COMP 550, Fall 2020

Due: **Wednesday, October 7th**, 2020, 9:00pm.

You must do this assignment individually. You may consult with other students orally, but may not take notes or share code, and you must complete the final submission on your own.

Sentiment Analysis

You will train models that classify a sentence into either a positive or negative sentiment. These sentences come from a movie review dataset constructed by the authors of this paper:

Bo Pang and Lillian Lee, Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales, *Proceedings of ACL 2005*.

The goal of this assignment is to give you experience in using existing tools for machine learning and natural language processing to solve a classification task. Before you attempt this assignment, you will need to install Python 3 on the machine you plan to work on, as well as the following Python packages and their dependencies:

- NLTK: `http://www.nltk.org/`
- NumPy: `http://www.numpy.org/`
- scikit-learn: `http://scikit-learn.org/stable/`

Download the corpus of text available in the attached file. This corpus is a collection of movie review sentences that are separated into positive and negative polarity. Your task is to train classifiers to distinguish them.

Data storage and format

The raw text files are stored in *rt-polarity.neg* for the negative cases, and *rt-polarity.pos* for the positive cases.

Research question

The main research questions being asked by your experiments will be: what machine learning model works well for sentence-level sentiment classification?

Preprocessing and feature extraction

Preprocess the input documents to extract feature vector representations of them. Your features should be unigram counts. You may also use scikit-learn's feature extraction module. You should experiment with whether to lemmatize or stem, and whether to include stopwords. NLTK includes implementations of lemmatizers and stemmers for English, as well as stopword lists. Also, remove infrequently occurring words as features. You may tune the threshold at which to remove infrequent words. You can also experiment with the amount of smoothing/regularization in training the models to achieve better results. Read scikit-learn's

documentation for more information on how to do this. There are many functions in these libraries that already implement many of the steps you will need for feature extraction.

Setting up the experiments

Design and implement an experiment that correctly compares the model variants, so that you can draw reasonable conclusions about which model is the best for generalizing to similar unseen data. This will require creating subsets of the dataset as we discussed in class. Compare the logistic regression, support vector machine (with a linear kernel), and Naive Bayes algorithms, each tuned for the preprocessing and feature extract decisions described above. In addition, compared against a fourth classification method of your choice, and against the expected performance of a random baseline, which just guesses positive or negative with equal probability. This fourth model can either be another classifier, or a different variant of one of the previous three.

Report

Write a *short* report on your method and results, carefully document i) the problem setup, ii) your experimental procedure (including the fourth model of your choice), iii) the range of parameter settings that you tried, and iv) the results and conclusions. It should be no more than 1.5 pages long. Report on the performance in terms of accuracy, and speculate on the successes and failures of the models. Which machine learning classifier produced the best performance? For the overall best performing model, include a confusion matrix as a form of error analysis.

Submitting code

Submit your code in a file named "a1.py".

What To Submit

Submit your report as a single pdf on myCourses called "a1-answers.pdf". In addition, you should submit one plaintext file with your source code called "a1.py". All work should be submitted to myCourses under the Assignment 1 folder.