ASSIGNMENT 1

COMP 550, Fall 2019

Due: **Saturday, September 28th**, 2019, 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.

Question 1:	40 points
Question 2:	15 points
Question 3:	45 points
	100 points total

Assignment

Question 1: Find Cases of Ambiguity (40 points)

Analyze the following passages by identifying the most salient instances of linguistic ambiguity that they exhibit. Write a *short* paragraph for each that answers the following questions. What is the ambiguity, and what are the different possible interpretations? What in the passage specifically causes this ambiguity, and what domain(s) of language does this cause involve (phonological, lexical, syntactic, orthographic, etc.)? What sort of knowledge is needed for a natural language understanding system to disambiguate the passage, whether the system is human or machine? Be more specific than simply saying "contextual knowledge."

1. *Every student took a course.*
2. *John was upset at Kevin but he didn't care.*
3. *Sara owns the newspaper.*
4. *He is my ex-father-in-law-to-be.*
5. *ttyl ;) [text message]*

Question 2: Naive Bayes and Logistic Regression (15 points)

Explain how a Naive Bayes classifier is a linear classifier over its feature space ϕ if we are using categorical distributions for the prior and the features by showing that there is an equivalent logistic regression model over the same feature space.

Question 3: Sentiment Analysis (45 points)

In this question, you will train a simple classifier that classifies 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 question 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 question, 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 a sentence classifier 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.

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.

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. Compare the logistic regression, support vector machine (with a linear kernel), and Naive Bayes algorithms. Also, compare against the expected performance of a random baseline, which just guesses positive or negative with equal probability.

Report

Write a *short* report on your method and results, carefully document i) the problem setup, ii) your experimental procedure, iii) the range of parameter settings that you tried, and iv) the results and conclusions. It should be no more than one page 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 "a1q3.py".

What To Submit

Submit your solutions to Questions 1 to 2, as well as the report part of Question 3 as a single pdf on myCourses called "a1-answers.pdf". For the programming part of Question 3, you should submit one plaintext file with your source code called "a1q3.py". All work should be submitted to myCourses under the Assignment 1 folder.