Assignment 2

There are two portions to this assignment: a practical portion and an analysis portion. For the practical portion, you will build a classifier using one or more of the machine learning algorithms covered in class. Once complete, you will answer the questions below based on the results you see from applying the algorithms against the data in the practical portion. You will submit your implementation code for the practical portion and your answers (as a text file or document in Word, OpenOffice or PDF) for the analysis portion.

Practical Portion

For the practical portion, you will use one or more of the machine learning algorithms, *Logistic Regression*, *Linear Discriminant Analysis / Quadratic Discriminant Analysis* or *k Nearest Neighbours*, against the data set provided. You may [read: should] use these algorithms as implemented in scikit-learn. Your implementation must take at least the following parameters:

- The name of the file for your training set
- The name of the file for your test set

The output of your implementation should look like:

```
Misclassification Rate = 0.4316
```

… and should be called from command line, as follows:

```
python my_script.py train.csv test.csv
```

You may also add any optional parameters as long as they are documented in your code and with run-time help. If the user fails to provide these parameters, the implementation may [read: should] print out the same run-time help before halting. Assuming the user provides all required parameters, the implementation should produce exactly one output: Misclassification Rate on the test set.

There is one data set for this assignment. This data set is proprietary. For legal reasons, you may not distribute this data, in part or whole, to anyone outside the University of San Francisco.

The data is in CSV format, representing a set of tweets. The fields are as follows:

Field	Description
1	One of {negative, neutral, positive}, unquoted, according to the perceived sentiment of the text in the tweet
2	Tweet content enclosed in double quotes

Your implementation must extract a number of features from the tweet (i.e. field 2) which predict the sentiment (field 1) of the tweet. The following features are required:

- Rates of the following English “function words”[1] in the tweet: {I, the, and, to, a, of, that, in, it, my, is, you, was, for, have, with, he, me, on, but}. For example, the rate of “the” in the phrase “the quick brown fox jumped over the lazy dog” is 2 / 9 = 0.2222.
- Rates of the following punctuation symbols in the tweet: {“.”, “,”, “!”}, defined as above.

Note that only the training set (train.csv) is provided. No test set will be provided to you; however, you must accommodate such a set in your implementation. You may [read: should] assume that the test set has the same format as the training set and resembles the training set while not being identical to it.

Analysis Portion

Based on the results of your implementation above, answer the following questions:

1. What, if any, features did you add to the implementation? What was your motivation?
2. In your opinion, which classifier(s) performed the best and why?
3. Were your additions helpful or harmful to the performance of your system?
4. What was your system’s improvement over the baseline, if any?
5. In what ways would you explore improving the performance of the models?

Grading

The implementation portion of this assignment is worth 80% of the grade, and the analysis portion the balance, evenly divided among the questions. The implementation is graded as follows:

- 40% = Proper implementation of the required features described above
- 5% = Implementation of each additional well-motivated change (addition to features, pruning of features, change to classifiers, etc.), to a maximum of 20%
- 5% = Improvement of 10% over baseline, to a maximum of 20%

Late assignments will receive 0%.

[1] For more, see Chung & Pennebaker (2007): The Psychological Functions of Function Words at ResearchGate or Google Books.