Homework 4

In this assignment, you will experiment with text classification: you will build a sentiment classifier that will predict whether a movie review is positive or negative. You will be working with movie review data, which can be downloaded from this site:

https://github.com/dennybritz/cnn-text-classification-tf/tree/master/data/rt-polaritydata

You will find two files there: one with positive and one with negative reviews. Your task is to build a binary classifier that will perform movie review classification automatically. You will need to implement (i.e. write from scratch) your own vectorizer to convert the text of the movie reviews into vectors that can be used to train a scikit-learn classifier. You will also implement (i.e. write from scratch) your own model selection procedure. Conretely, here are the steps that you need to take (each is worth 20 points):

1. Randomly split the data into training (80%) and test (20%) sets. Keep the test set away in a separate location until you are done with model selection.
2. Implement your own vectorizer that converts the training data into a numpy array of shape (num_of_training_examples x num_of_features). To control the dimensionality of the resulting vectors, you can discard the features that occur in fewer than n examples and/or more than m examples. You will pick the actual value of n (e.g. 5) and/or m (e.g. 5000) in the process of model selection. Note: you are not allowed to use third-party implementations such as scikit-learn's CountVectorizer etc.
3. Train a classifier of your choice (e.g. logistic regression, SVM) using n-fold cross validation. You are required to implement your own n-fold cross-validation rather than using the scikit-learn API. I.e. you will need to randomly split the training set into n parts and repeatedly train on n-1 parts and test on the remaining part as we discussed in class.
4. Implement your own grid search procedure which should include a search over at least two hyper-parameters one of which could be the parameter that controls the dimensionality of your data and the other one is a classifier-specific hyper-parameter such as C in logistic regression and SVM.
5. Use the vectorizer you created in step 2 to convert the test set into a numpy array of shape: (num_of_test_examples x num_of_features). Pick the best model and report its performance on the test set. Think about what performance metric is appropriate for this data and justify your choice.

The ability to summarize your findings is extremely important when doing scientific research or working as a data scientist. Please summarize your findings

in a (up to) 1-page paper. Include the details on what kind of pre-processing you performed and your choice of model hyper-parameters. This assignments will be evaluated primarily based on your writeup. Please see the general guidelines for homework submission in the syllabus.