Using nltk for Sentiment Analysis

nltk – suite of libraries and programs for NLP.

Project Plan:

The steps I followed for the task-

- Getting the data from nltk.corpus.
- Tokenize the tweets into words .
- Import English stop words and make a set of emoticons.
- Remove the stopwords, handles, hyperlinks, hashtags and emoticons from the tweets.
- Use bag_of_words to vectorize the words and (1 or 0 based on presence, after checking the vocabulary).
- Bag of words returns a dict with the word as key and value as true or false .
- Shuffle positive_tweets.json and negative_tweets.json randomly and take 1:4 split to make the test and train sets.
- Use NaiveBayesClassifier from nltk.classify to train on the training set
- Check accuracy on test set and then check the performance on a test example print out the confidence of the model on the specific example.
- Create defaultdicts with default value as an empty set for the actual and estimated sentiments of the texts to get an empty set for a label which is missing in the actual_sets or predicted_set.
- Create ConfusionMatrices to visualize predicted and actual positives and negatives.

List of things I want to add but I didn’t have time to learn:

- Using SVM Classifiers from sklearn and adjusting the parameters given to the classifier.
- Getting a bigger dataset from Kaggle.
- Using GridSearchCV algo from sklearn to get better weights for the estimator.
- Using tweepy module to stream tweets from twitter app.
- Using RNN to solve the given task.