# Summary of alternative model selection and results

In this part of the project we used the NLTK library results, that we achieved in the previous part, as the ground truth. This time, instead of feeding in the tweet as text we used by bag-of-words model to first convert the tweet into n-dimensional vector space, where ‘n’ was determined using frequency of the most important words. We should state that while bag-of-words has been superseded by Word2Vec models, it still holds the upper hand in cases where the amount of data and the variation within data is less. In our case, because twitter has a limit on the number of characters, and we only deal with data related to politics, BOF models are likely to work better.

**Bag of Words**

BOF is a simplified representation of text data that selects frequency over grammar and word order. The underlying reasoning behind the model is that words that have higher impact on the sentiment are unique in sentences and extracting those words will yield a better accuracy.

There are two parts to creating this model, Tokenizing and Lemmatizing. Tokenizing refers to breaking up a sentence to create individual tokens of each word and lemmatizing allows us to convert multiple forms of same word to a common form (ran, run, running -> run). Once we have our corpus, we create a lexicon which contains frequently used words.

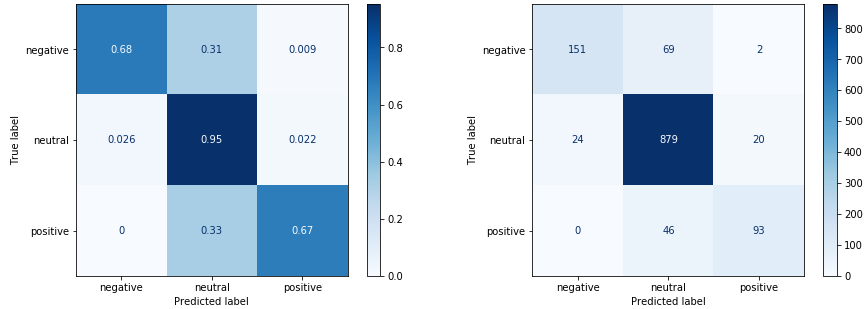
Using the lexicon, we then convert each tweet to a vector space of size n, where ‘n’ is the number of words in the generated lexicon. If a tweet has a word that is in the lexicon, then we increment the counter by 1 at the position of the word in the lexicon.

**Models**

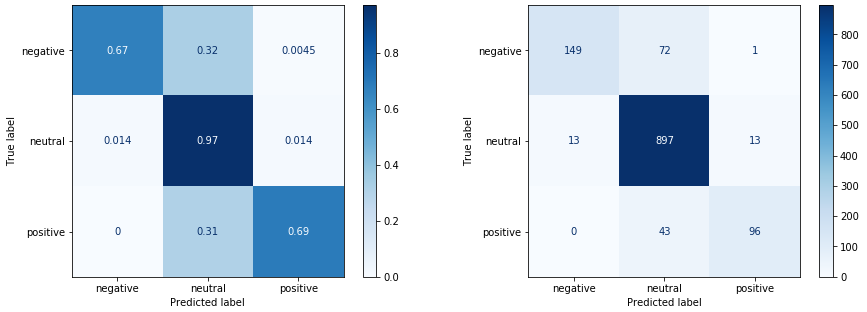
Using this, we converted all our tweets into 459 dimensional vectors containing mostly 0s and a few 1s. We then used scikit-learn to train 4 different models using our previous model output as the ground truth.

**Training Data: 5185 (pos: 823, neg: 598), Testing Data: 1284 (pos: 222, neg:139)**

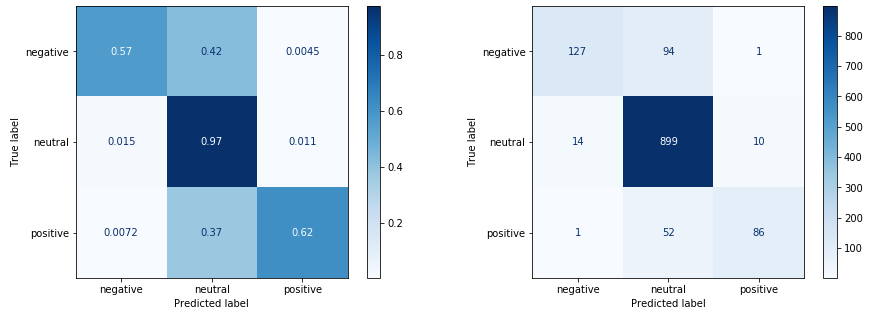
1. Random Forest Classifier



1. Linear SVM



1. Adaboost Classifier



1. MLP

