Jude Kappel
NLP Assignment 1 Breakdown
Prof. Luo
Fordham University

NaiveBayesClassifier.py:

1. Instance Variables:
    a. Corpus: set, stores unique words (vocabulary).
    b. Word Frequencies: defaultdict(int), counts occurrences of each word per class.
    c. Class Prior Probability: dict, holds the probability of each class.
    d. Class Word Counts: dict, tracks total words per class.
    e. Stopwords: set, contains words to ignore during tokenization.

2. Cleaning and Tokenization
    a. Lowercase tweet, remove special characters, append to a list of tokens if word is not in stopword set

3. Fitting Model
    a. calculates class prior probability
    b. Calculates word counts
    c. Tokenize tweet
        i. Add token to corpus
        ii. Calculate frequency of word given class
    d. Update class prior probability

4. Predict
    a. For tweet in X_test
        i. Calculate the logged probability of the tweet belonging to each class
        ii. Calculate likelihood of tweet (with smoothing)
        iii. Append prediction to list
        iv. Return prediction list

NaiveBayes.ipynb:

1. Download all the NLTK data (twitter samples, stopword repo)

2. Concatenate positive and negative tweets together and map them to the corresponding class labels

3. Undersample the data (800 training examples, 200 validation examples) without replacement and ensuring the indices of the training data do not appear in the validation data

4. Fit the model to the training data and predict on test instances

5. Evaluate the model with performance metrics
    a. Accuracy, precision, recall, f1 score