For the preprocessing of these documents, we first extracted all the files from each of the document folders. We then created a DTM matrix for each of the authors, and cleaned up the vocabulary by removing stop words and making all words lowercase, as well as removing all punctuation and the extra white space. Additionally, we calculated the TFIDF for each term, and then added the author name to each of the rows for classification purposes, after converting to a dataframe format. We then created one dataset with the terms in all the files, with 2500 rows, 50 documents per author. This was repeated for both the training and testing.

Afterwards, we calculated the cosine distance between each of the terms and created a cosine distance matrix for both the training and testing sets. To deal with discrepancies in the number of rows (word differences), pseudo counts were added. After calculating the cosine distance between pairs of words, we fit a KNN model on the training data, using k= 50, which returned to us a prediction accuracy on the testing set of about 2%. We then changed the k value to k = 70, which gave us the exact same prediction accuracy. While there were about 50 documents with the author correctly predicted, in the grand scheme of 2500 documents overall in the test set, it seems a little low.

As such, we then tried one more model- the Naive Bayes Classifier. Since R-studio doesn't allow for Bayes to run without having the same column names, we removed all the columns that had words that were not in both datasets, which does induce lower accuracy, but we could not figure out how to assign an arbitrary value for words that did not show up in the training data. The Naive Bayes classifier gave us an accuracy of around 22.3%, which is much better than KNN and thus we choose Naive Bayes as the model we use in order to accurately classify documents.