Cuisine Type Classifier

To create a model for classifying cuisine types from this data, the first thing we must decide is the type of labels we want to include as part of a classifier. If this online recipe publisher generally gets its web traffic from the United States of America, then it is important to focus on cuisine types that are popular there as well. For example, some of the most popular cuisine types in the USA besides American are Mexican, Italian, Indian, Chinese, Japanese, and Cuban; but there are other types of cuisines that are worth noting such as Thai, Greek, and French. Having too many labels to choose from will overcomplicate the model making it not that effective or accurate. To prevent this from happening, we will limit the total number of categories to the ten I have listed above.

When training the model, just like the text classifier models I have made above, the best approach would be to use a semi-supervised technique. This would mean having to manually label the text data from the examples given to use. Obviously, many of these cuisines have dishes or ingredients that are only specific to them, meaning training the model to look for those specific words/n-grams would be the best way to increase accuracy of the label. The difficulty arises from when cuisines have overlap and share their dishes or ingredients (which many of them do). For example, both Chinese and Japanese cuisines share a love for rice and seafood. Therefore, it is important to have labeled examples that can specifically distinguish these two cuisines based on the name of the dish, the techniques used for the recipe, equipment used to cook with, or other specific ingredients used alongside hard to label ones.

Ultimately, the steps to train and find the best model to classify cuisine types is not too different from the steps I took to create the relevant text classifier. The main obstacle is determining the useful labels and how the model can identify tricky scenarios for them.