Task 3C: Analyze a less obvious dataset:SmS Spam Filtering

There is a given dataset contains 5574 text messages from UK labeled either spam or ham.
Modeling techniques:
After have a view of the dataset, we found all the data is regular text so we can use natural language processing techniques to filter messages.

Data transformation:
There are two data transformations that we can use on the dataset.

The first one is CountVectoizer:
It is a way of extracting features from the text for use in machine learning algorithms. The process of it is converting a collection of text documents to a matrix of token counts by counting the number of times each word appears in a document.

The second one is TFIDF vector:
TFIDF is short for term frequency–inverse document frequency. TF is word frequency which means how many times a word appears in a document and IDF reflects how often a word appears in all text. If a word appears in many texts, its IDF value should be low such as some stopwords ‘the’ or ‘a’.

In our project we first choose TF IDF vector and use it as features to train the dataset. We aim to get a predictor which can help us tell a mail belongs to ham or spam. Therefor we need to perform text mining on the given dataset, fit a predictive model on top of that and suggest improvements to increase our proposed model’s performance.

Model:
We will use Naive Bayes model for prediction.
There are several approaches:

1) data preprocessing: Before the data transformation, we should preprocess the data. First we change all the upper letter into lower letter and then we remove the stopwords from it.
2) data transformation: We transform all the text data into vectors. In this step, we use both CountVector and TFIDF vector.
3) split data set into train and test: We use train_test_split tool in sklearn to divide the data into train and test and set seed=100 to get more data.
4) data prediction: Using Naive Bayes model to predict it.
   Here is the theory about it: We set Y as the result of mail prediction(spam or ham), and X as the words in the mail. P (Y) represents the probability of each type of email in the training set. P (X) represents the probability of a certain

word in the email, and P (X | Y) represents the probability of the word X in a certain type of email. P (Y | X) represents the probability that the message is in category Y when the message contains X, which is the result of spam filtering. We will classify the test sample X into the category Y that maximizes P (Y | X).

Then we calculate the accuracy of the prediction:

TFIDF: 0.961

Improve the result:

Then we try to use CountVectoizer as the data transfomation method. And then we get the accuracy of prediction:

CountVectoizer: 0.989

We can see that there is an improvement of our result.

Conclusion:

We try to classify ham and spam using some natural language processing tools and Naive Bayes algorithm. We obtain 96% accuracy using TF IDF vector and then we use CountVectorizer as data transformation method and get 99% accuracy.