Assignment 1

Question 1 Input and Basic pre-processing

As this classifier is only using “REAL” or “FAKE” first thing I did was complete the function parse_data_line, converting the label and separate the statement from the data line and populating these 2 values in new separate label and statement variables.

Cleaning and pre-processing data are important steps prior to building, training, and testing a model. In this assignment I did this by separating the punctuation from the beginning and ends of the strings this allows for identical words to be weighted and treated equally, I then split the text on the whitespace remaining following this I separated the text and normalised the text to lower case lower casing makes casing equal throughout although choosing where to do this can affect the result.

Question 2 Basic feature extraction

The next part focused on extracting features from the data, to do this I imported the CountVectorizer library which provides frequency feature functionality. I then created an empty global feature dictionary which will be populated for increased efficiency, this dictionary holds the tokens and corresponding weights. Next I built the function to take tokens and iterate through updating the weights based on whether the token is already in the dictionary or not, I held the results in a temporary dictionary and used this to update the existing dictionary which will allow the text to be used for the modelling.

Question 3 Cross Validation

The next function I completed for use in building the model is the cross validate function, the function operates by taking variying sizes of the dataset to test and train the model. Firstly I imported the classification_report library from sklearn as I’d be using this in the function. Next I move onto the function, The function takes 2 variables, the dataset and the folds. The fold size is assigned and then I iterate through the dataset.

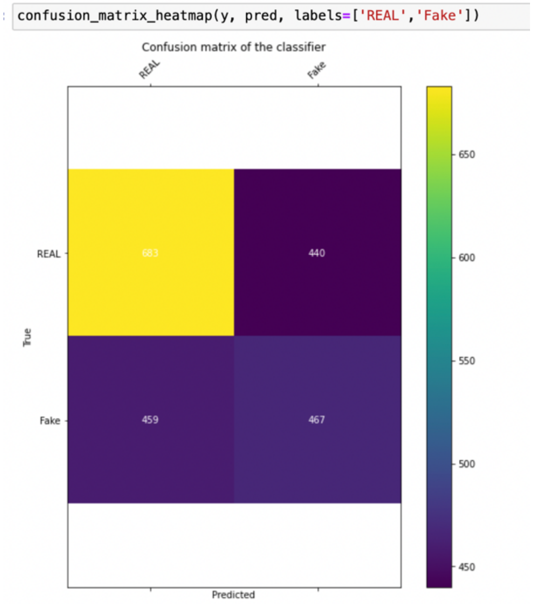
Inside of the iteration I separate out the training and testing samples which will change in size on each iteration, I then train the model on the training sample and check the models success against the testing sample returning the results at the end of the function for analysis.
The data is then loaded, split and pre-processed ready for training where the data is then trained and then tested. Then we call the function to train the classifier.

Question 4 Error Analysis

Next I look into the error analysis, here I use the confusion heatmap to display the predictions against the outcomes.
The heatmap shows the model and the lack of quality in predicting ‘FAKE’ labels, you can see in the heatmap that true positives are relatively well predicted however amount of false positives and false negatives in comparison to the amount of true negatives.

I believe the pre-processing of text affects the quality of classifier which is a potential issue as the test is taken out of context which can hinder the quality of the classifier but is also a necessary step in assisting and making an efficient classifier aiding the modelling of the classifier.

I analysed the first fold and added the false negatives to a file and the false positives to a file, incorrect words associated with labels adversely affected the models accuracy which is reinforced by this.



```
confusion_matrix_heatmap(y, pred, labels=['REAL','Fake'])
```

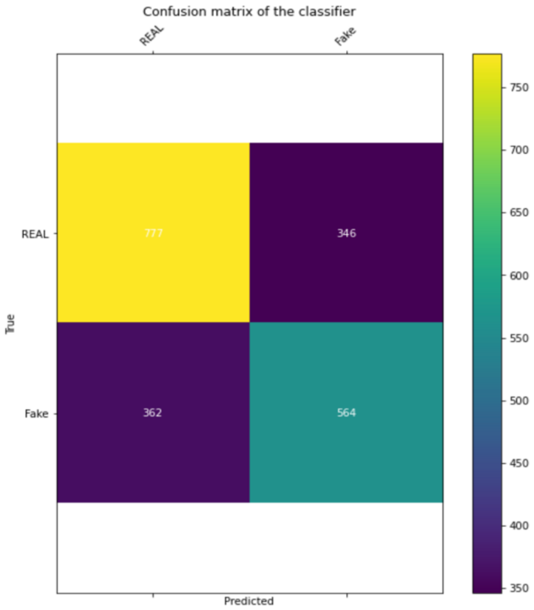
Question 5 Optimising pre-processing and feature extraction

Once the training model had been complete I sampled different pre-processing measures to see how this affected the model. The initial model I built already removed punctuations, so I continued by adding stemming which had an adverse effect on the punctuation, I also trialled stemming and lemmatising but these functions also had an adverse effect on the classifier reducing the accuracy as seen in the table below.

Pre-Processing style	Accuracy
Remove punctuation	62%
Remove punctuation with stemming	59%
Remove punctuation with lemmatising	61%
Stop words and remove punctuation	60%

Question 6 Using other metadata in the file

To improve the classifier, I introduced other metadata into the file, introducing columns 3 – 7 seemingly had no supporting affect on the classifier and only increased with column 8 to a 64% this was the single highest increase of an added column, 9, 10, 11 and 12 also had a positive effects on the accuracy when included in the file however not as high singularly as 8 so I coupled in 8, 9 and 10 with column 8 increasing the accuracy to 68%. I then added the existing columns and introduced 11 and 12 I didn't add 13 as it had no singular effect, this increased the accuracy to 71%. The below heatmap shows my final with the data columns 8,9,10,11,12 added and punctuation removed from the beginning and end of strings and tokenisation.



Confusion matrix of the classifier
REAL
Fake
REAL
Fake
True
Predicted