Assignment 1

STEP 2: Familiarise:

2. Please find the screenshot of the code which calculates the metrics for Test Bayes

```python
# TODO for Step 2: Add some code here to calculate and print: (1) accuracy; (2) precision and recall for the positive class;
# (3) precision and recall for the negative class; (4) F1 score;
def calculateMetricsForTestBayes(correct, total, correctpos, totalpospred, totalpos, correctneg, totalnegpred, totalneg,
dataName):

        print("\nMetrics for: ", dataName)
        print("Accuracy: ", correct/total)

        precision_pos = correctpos/totalpospred
        recall_pos = correctpos/totalpos
        f1_score_pos = (2 * precision_pos * recall_pos) / (precision_pos + recall_pos)
        print("Precision for the positive class: ", precision_pos)
        print("Recall for the positive class: ", recall_pos)
        print("F-Measure for the positive class: ",  f1_score_pos)

        precision_neg = correctneg/totalnegpred
        recall_neg = correctneg/totalneg
        f1_score_neg = (2 * precision_neg * recall_neg) / (precision_neg + recall_neg)
        print("Precision for the negative class: ", precision_neg)
        print("Recall for the negative class: ", recall_neg)
        print("F-Measure for the negative class: ",  f1_score_neg)
```

3. Classification result for the test set is as follows:

```
Metrics for:  Films  (Test Data, Naive Bayes)
Accuracy:  0.7764265668849392
Precision for the positive class:  0.7757009345794392
Recall for the positive class:  0.7771535580524345
F-Measure for the positive class:  0.7764265668849392
Precision for the negative class:  0.7771535580524345
Recall for the negative class:  0.7757009345794392
F-Measure for the negative class:  0.7764265668849392
```

The accuracy of the test data is about 77.64% and the recall for the positive and the negative class is 77.71% and 77.57% respectively.

STEP 3: Run Naive Bayes on other data:

1. Naive Bayes on films (Train Data)
   Accuracy: 88.92%

```
Metrics for:  Films (Train Data, Naive Bayes)
Accuracy:  0.8892015843235356
Precision for the positive class:  0.8955508474576271
Recall for the positive class:  0.8811757348342715
F-Measure for the positive class:  0.8883051381737942
Precision for the negative class:  0.8830529339351662
Recall for the negative class:  0.8972274338127997
F-Measure for the negative class:  0.8900837555578535
```

Naive Bayes on Nokia (All Data)
Accuracy: 60.90%

```
Metrics for:  Nokia   (All Data,  Naive Bayes)
Accuracy:  0.6090225563909775
Precision for the positive class:  0.8014705882352942
Recall for the positive class:  0.5860215053763441
F-Measure for the positive class:  0.6770186335403726
Precision for the negative class:  0.4076923076923077
Recall for the negative class:  0.6625
F-Measure for the negative class:  0.5047619047619047
```

2. It is observed that the accuracy for the Films train and the Films test data for Naive Bayes approach is comparatively higher than that of Nokia (All Data). The reason being the corpus used to feed the Naive Bayes classifier was Films train data, which was then used to predict the class for all the three data sets. Nokia (All Data) wasn't trained and hence we could see the result classified poorly. Naive Bayes approach calculates the Likelihoods and uses them to predict the final decision. Since the likelihoods for all the words in training set corpus were calculated, the predictions have an accuracy of approximately 88%. Another reason for the difference in the results is the domain of both the data sets one being movie reviews and the other being phone reviews which are completely different.

STEP 4: What is the model being learnt?

The following are the top 50 most useful words printed by the function.

```
NEGATIVE:
['unfunny', 'mediocre', 'routine', 'badly', 'generic', 'pointless', 'poorly', 'boring', 'bore', 'mindless', 'bears', 'stale
', 'annoying', 'tiresome', 'unless', 'apparently', 'save', 'offensive', 'disguise', 'stupid', 'meandering', 'product', 'plo
dding', 'pinocchio', 'retread', 'animal', 'horrible', 'harvard', 'overwrought', 'plotting', 'shoot', 'waste', 'literally',
'stealing', 'banal', 'chan', 'fatal', 'incoherent', 'seagal', 'dull', 'supposed', 'junk', 'bother', 'inept', 'cable', 'amat
eurish', 'kung', 'pathetic', 'trite', 'unintentional']

POSITIVE:
['answers', 'literary', 'hopeful', 'culture', 'unexpected', 'evocative', 'iranian', 'captivating', 'spare', 'timely', 'poig
nant', 'vividly', 'record', 'unfolds', 'smarter', 'warm', 'powerful', 'pulls', 'playful', 'tour', 'delightful', 'polished',
 'depiction', 'jealousy', 'heartwarming', 'bittersweet', 'wry', 'touching', 'beauty', 'captures', 'tender', 'intense', 'liv
ely', "world's", 'provides', 'detailed', 'chilling', 'wonderfully', 'thoughtful', 'realistic', 'delicate', 'gem', 'mesmeriz
ing', 'refreshingly', 'inventive', 'intimate', 'riveting', 'refreshing', 'engrossing', 'wonderful']
Count of Negative words in sentimentDictionary: 23
Count of Positive words in sentimentDictionary: 29
```

The words selected here are based on the corpus of the Films (Train Data) and hence are good words for predicting the sentiment. These sentiments are certainly good for the Films data set, but not the Nokia data set. For instance, words like unfunny, mediocre, routine in Negative wordlist and captivating, hopeful, heart-warming in the positive wordlist are good sentiments for film data set and not the mobile data set.
As seen in the screenshot, there are 23 negative words and 29 positive words in the sentiment dictionary.

STEP 5: How does a rule-based system compare?

1. The metrics for Films (Train), Films (Test) and Nokia data set is as follows:

```
Metrics for:  Films (Train Data, Rule-Based)
Accuracy:  0.5456240290005179
Precision for the positive class:  0.5402861445783133
Recall for the positive class:  0.5960539979231568
F-Measure for the positive class:  0.5668016194331984
Precision for the negative class:  0.5521528897075754
Recall for the negative class:  0.4954545454545455
F-Measure for the negative class:  0.5222694108679081

Metrics for:  Films  (Test Data, Rule-Based)
Accuracy:  0.5357142857142857
Precision for the positive class:  0.5415224913494809
Recall for the positive class:  0.6065891472868217
F-Measure for the positive class:  0.5722120658135283
Precision for the negative class:  0.5279069767441861
Recall for the negative class:  0.4613821138211382
F-Measure for the negative class:  0.4924078091106291

Metrics for:  Nokia   (All Data, Rule-Based)
Accuracy:  0.6766917293233082
Precision for the positive class:  0.7941176470588235
Recall for the positive class:  0.7258064516129032
F-Measure for the positive class:  0.7584269662921348
Precision for the negative class:  0.46875
Recall for the negative class:  0.5625
F-Measure for the negative class:  0.5113636363636364
```

2. The dictionary-based approach performed poorly for Films (Train and Test Data) compared to the Naive Bayes approach as the accuracy of the model has decreased from 88.92% and 77.64% to 54.56% and 53.57% respectively for the train and test data respectively. The Nokia data set seems to have performed good for the dictionary-based approach compared to the Naive Bayes and the accuracy has increased to 67.66% from 60.90%. This is because *sentimentDictionary* is used to calculate the final scores and there are a greater number of common words in the sentiment dictionary based on the mobile dataset compared to Films dataset.

3. A new function *testDictionaryModified* has been written which improves the performance for all the data sets. Here we tweak the weights of the sentiment dictionary by taking into account *mostUseful* wordlist for positive and negative words. We modify the weight for the most negative words and most positive words in the wordlist to -2 and 2 respectively and store it in the *sentimentDictionaryModified.* Also, the unwanted words picked up from the positive and negative word text file have been omitted. For each word in the sentence, if the word is in the positive dictionary, it adds 1, if it is in the negative dictionary, it subtracts 1. If the word is in the most useful positive words it adds 2, if in most useful negative words, it subtracts -2. Once the final score is calculated for the sentence, a check for any negation word is carried out, if encountered, then the polarity of the score is reverted.

Following are the results for rule based for all data sets after code modified for accuracy:

```
Metrics for:  Films (Train Data, Rule-Based)
Accuracy:  0.6406007250129466
Precision for the positive class:  0.6421475375184951
Recall for the positive class:  0.630944963655244
F-Measure for the positive class:  0.6364969620783575
Precision for the negative class:  0.6391145410235581
Recall for the negative class:  0.650206611570248
F-Measure for the negative class:  0.6446128635804999

Metrics for:  Films  (Test Data, Rule-Based)
Accuracy:  0.6378968253968254
Precision for the positive class:  0.6513026052104208
Recall for the positive class:  0.6298449612403101
F-Measure for the positive class:  0.6403940886699506
Precision for the negative class:  0.6247544204322201
Recall for the negative class:  0.6463414634146342
F-Measure for the negative class:  0.6353646353646354

Metrics for:  Nokia   (All Data, Rule-Based)
Accuracy:  0.7556390977443609
Precision for the positive class:  0.8622754491017964
Recall for the positive class:  0.7741935483870968
F-Measure for the positive class:  0.8158640226628896
Precision for the negative class:  0.5757575757575758
Recall for the negative class:  0.7125
F-Measure for the negative class:  0.6368715083798883
```

STEP 6: Error Analysis.

a. The score is calculated based on a set of positive words and negative words and not related to the data set.
b. The *sentimentdictionary* generated includes even the commented lines on top of both positive-words.txt and negative-words.txt files, due to which the accurate positive and negative words are not been identified by the classifier and negative weights are given to all the additional words present in the commented words included in the dictionary.
c. Negation handling need to be handled by the classifier. Also, handling negative and positive words in combination with those negation words need to be taken into consideration.
   For example, It *isn't a bad* phone.
   (Negation) + Negative words = Positive sentiment

Please find the snapshot of few errors displayed for the *testDictionary Nokia data set.*

```
ERROR (pos classed as neg -5.00):there is much which has been said in other reviews about the features of this phone
, it is a great phone , mine worked without any problems right out of the box .
ERROR (pos classed as neg -9.00):i have had the phone for 1 week , the signal quality has been great in the detroit a
rea ( suburbs ) and in my recent road trip between detroit and northern kentucky ( cincinnati ) i experienced perfect
signal and reception along i-75 , far superior to at &#38; t 's which does not work along several long stretches on t
hat same route .
ERROR (pos classed as neg -6.00):my favorite features , although there are many , are the speaker phone , the radio a
nd the infrared .
ERROR (pos classed as neg -5.00):the speaker phone is very functional and i use it in the car , very audible even wit
h freeway noise .
ERROR (pos classed as neg -6.00):the infrared is a blessing if you have a previous nokia and want to transfer your ol
d phone book to this phone , saved me hours of re-entering my numbers .
ERROR (pos classed as neg -4.00):it has lots of little cute features , my favorite being the games and the pim ( pers
onal information manager -- i.e. organizer ) , and the radio !
```