CMPE 493

Introduction to Information Retrieval

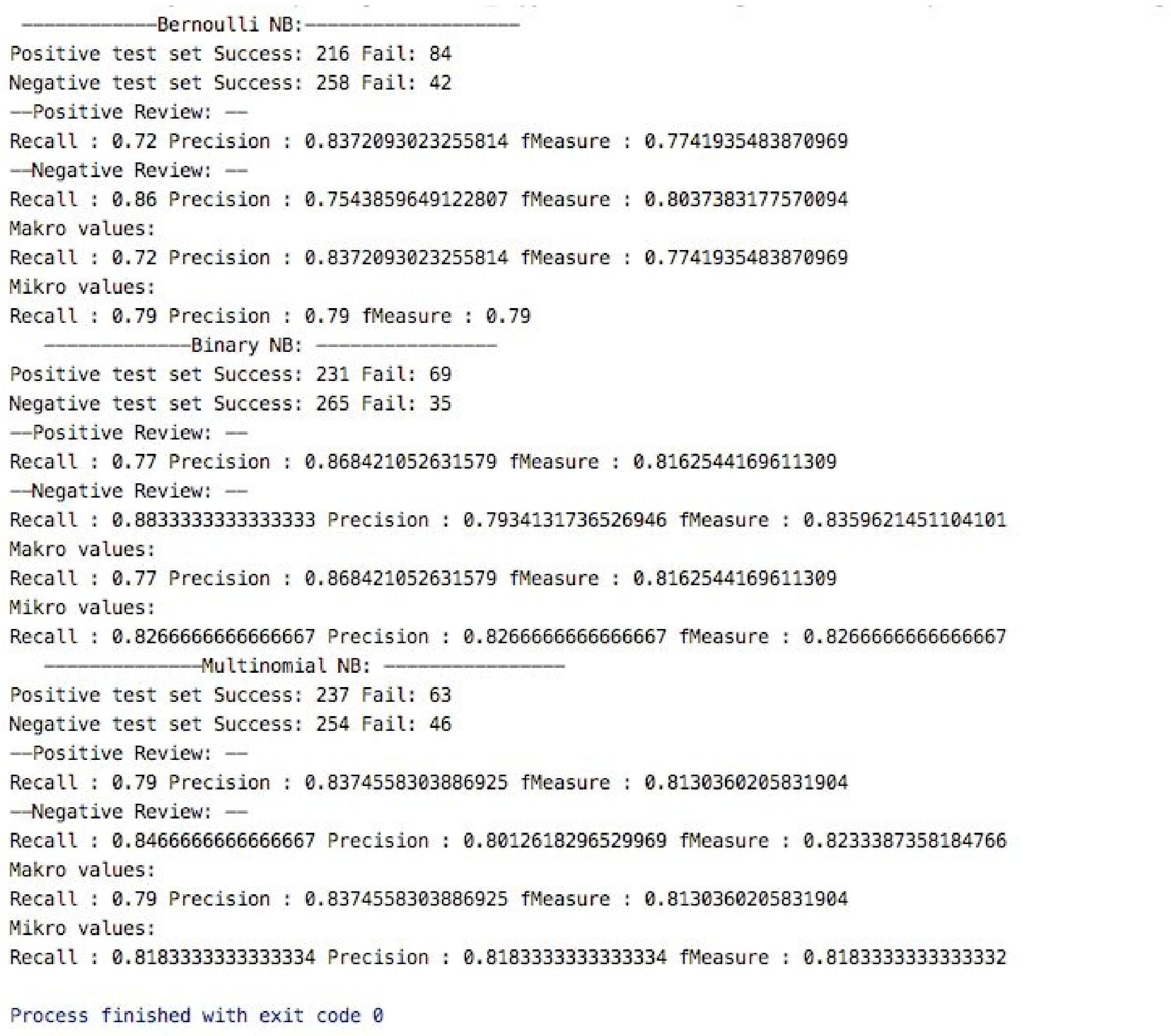
Movie Review Sentiment Classification

REPORT

Şadi Uysal

2015400162

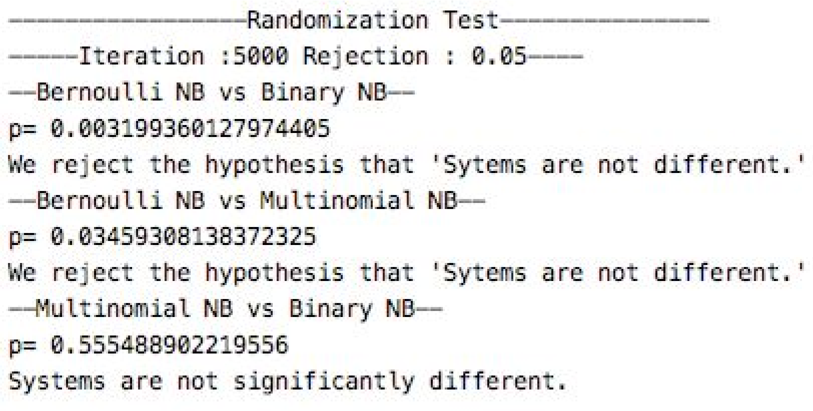
a)Report the *macro-averaged* and *micro-averaged* precision, recall, and F-measure values obtained by your classifiers on the test set, as well as the performance values obtained for *each class separately* by using *Laplace smoothing* with α = 1



```
 -------------Bernoulli NB:-------------------
Positive test set Success: 216 Fail: 84
Negative test set Success: 258 Fail: 42
--Positive Review: --
Recall : 0.72 Precision : 0.8372093023255814 fMeasure : 0.7741935483870969
--Negative Review: --
Recall : 0.86 Precision : 0.7543859649122807 fMeasure : 0.8037383177570094
Makro values:
Recall : 0.72 Precision : 0.8372093023255814 fMeasure : 0.7741935483870969
Mikro values:
Recall : 0.79 Precision : 0.79 fMeasure : 0.79
   -------------Binary NB: ---------------
Positive test set Success: 231 Fail: 69
Negative test set Success: 265 Fail: 35
--Positive Review: --
Recall : 0.77 Precision : 0.868421052631579 fMeasure : 0.8162544169611309
--Negative Review: --
Recall : 0.8833333333333333 Precision : 0.7934131736526946 fMeasure : 0.8359621451104101
Makro values:
Recall : 0.77 Precision : 0.868421052631579 fMeasure : 0.8162544169611309
Mikro values:
Recall : 0.8266666666666667 Precision : 0.8266666666666667 fMeasure : 0.8266666666666667
   --------------Multinomial NB: --------------
Positive test set Success: 237 Fail: 63
Negative test set Success: 254 Fail: 46
--Positive Review: --
Recall : 0.79 Precision : 0.8374558303886925 fMeasure : 0.8130360205831904
--Negative Review: --
Recall : 0.8466666666666667 Precision : 0.8012618296529969 fMeasure : 0.8233387358184766
Makro values:
Recall : 0.79 Precision : 0.8374558303886925 fMeasure : 0.8130360205831904
Mikro values:
Recall : 0.8183333333333334 Precision : 0.8183333333333334 fMeasure : 0.8183333333333332

Process finished with exit code 0
```

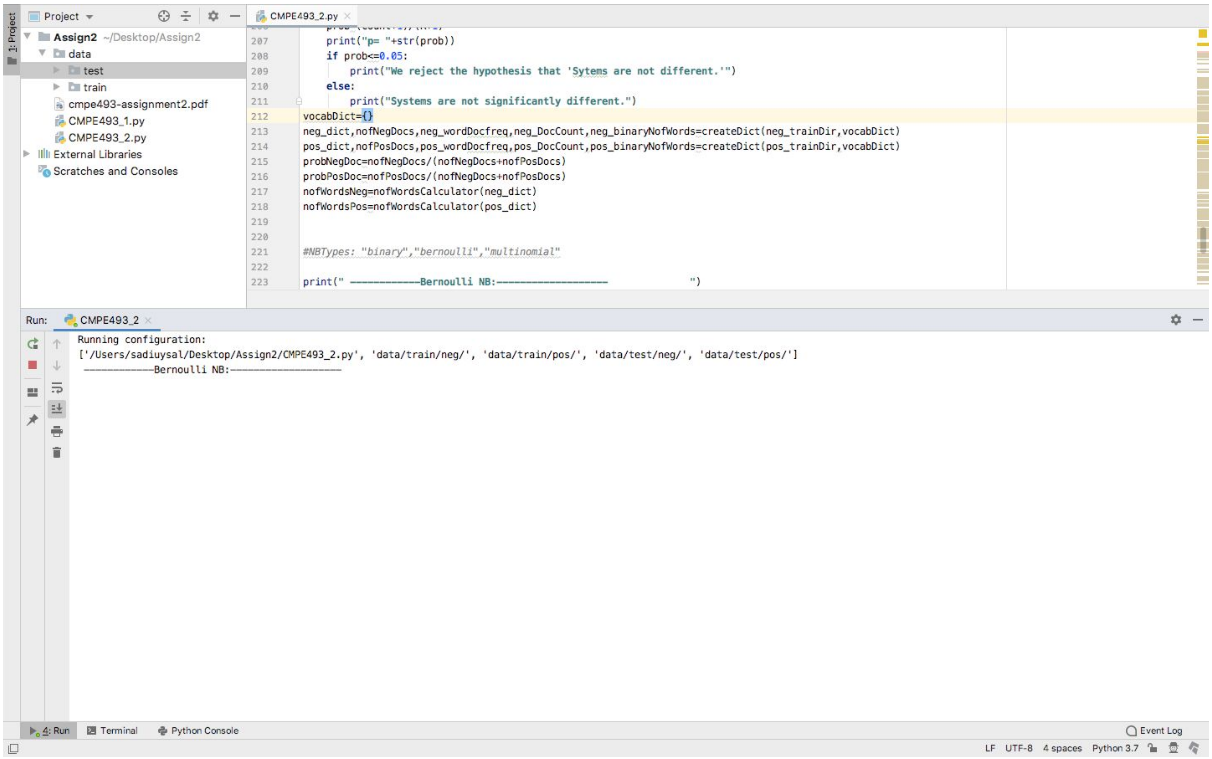
b) Compare and discuss the performance of each NB model for this task. Perform randomization tests to measure the significance of the differences between the micro-averaged F-scores of the algorithms.



```
-----------------Randomization Test---------------
-----Iteration :5000 Rejection : 0.05----
--Bernoulli NB vs Binary NB--
p= 0.003199360127974405
We reject the hypothesis that 'Sytems are not different.'
--Bernoulli NB vs Multinomial NB--
p= 0.03459308138372325
We reject the hypothesis that 'Sytems are not different.'
--Multinomial NB vs Binary NB--
p= 0.555488902219556
Systems are not significantly different.
```

Bernoulli NB model is especially popular for classifying short texts. Because of the benefit of explicitly modelling the absence of terms. But in our case, this modelling is not better performed than others. I thought the reason for that would be the overall length of reviews. Moreover, best performed modelling is Binary NB model. It shows that in our case, word frequencies in the document gives not that much information while categorizing reviews.

c) Include a screenshot showing a sample run of your program.



```python
    print("p= "+str(prob))
    if prob<=0.05:
        print("We reject the hypothesis that 'Sytems are not different.'")
    else:
        print("Systems are not significantly different.")
vocabDict={}
neg_dict,nofNegDocs,neg_wordDocfreq,neg_DocCount,neg_binaryNofWords=createDict(neg_trainDir,vocabDict)
pos_dict,nofPosDocs,pos_wordDocfreq,pos_DocCount,pos_binaryNofWords=createDict(pos_trainDir,vocabDict)
probNegDoc=nofNegDocs/(nofNegDocs+nofPosDocs)
probPosDoc=nofPosDocs/(nofNegDocs+nofPosDocs)
nofWordsNeg=nofWordsCalculator(neg_dict)
nofWordsPos=nofWordsCalculator(pos_dict)


#NBTypes: "binary","bernoulli","multinomial"

print(" ------------Bernoulli NB:-------------------             ")
```

```
Running configuration:
['/Users/sadiuysal/Desktop/Assign2/CMPE493_2.py', 'data/train/neg/', 'data/train/pos/', 'data/test/neg/', 'data/test/pos/']
 ------------Bernoulli NB:-------------------
```