# <u>Analysis</u>

## Shalini Bhardwaj(MT19045)

Naïve Bays: Uses conditional probability to classify documents to respective class.It sees the terms of the class and then assign score to each class.Highest Class is labelled on testing doc.

I tried variation of split and applied both tfidf and mutual information on naive bayes and analysed.

As the documents were randomly shuffled each time variations were seen in the results.

Below are attached confusion matrix and accuracy score  with each mI and tfidf as feature selection.

As I took k as around 10,000 unique words that is 50% of features from each class. K is chosen such that top n is normalized with respect to the length of terms in class also.

It was seen that  in naïve bays nutual information was performing better and nearly 90+ accuracy are achieved from both the methods.

As we were increasing the traing size from 50 to 80 better accuracy was achived and is shown in graph below.

For mutual information Smoothening is done so that if term was not present in traing than that document should not be nullified due to that word.

Here tf =term frequency I that class

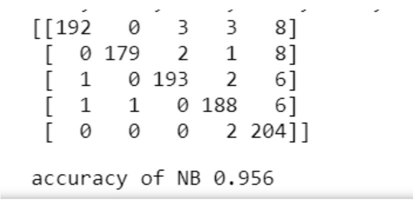
Idf= number of class in which term ispresent (in detailed explained in readme)

Naïve Bayes

80 :20

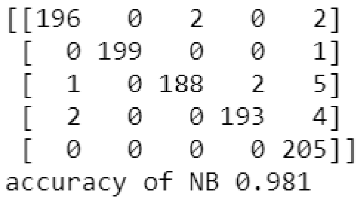
Tfidf

MI



```
[[192   0   3   3   8]
 [  0 179   2   1   8]
 [  1   0 193   2   6]
 [  1   1   0 188   6]
 [  0   0   0   2 204]]

accuracy of NB 0.956
```

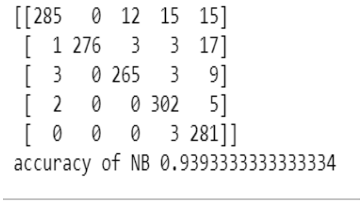


```
[[196   0   2   0   2]
 [  0 199   0   0   1]
 [  1   0 188   2   5]
 [  2   0   0 193   4]
 [  0   0   0   0 205]]
accuracy of NB 0.981
```

Here MI is performing better than  tfidf and it’s the best split of all .As it considers the mutual information of class c1 with a term of that class it also considers presence of that term in other classes.

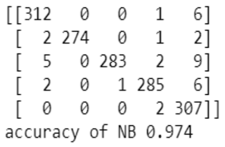
70:30

Tfidf



```
[[285   0  12  15  15]
 [  1 276   3   3  17]
 [  3   0 265   3   9]
 [  2   0   0 302   5]
 [  0   0   0   3 281]]
accuracy of NB 0.9393333333333334
```

MI

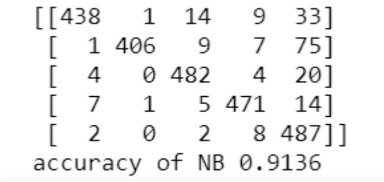


```
[[312   0   0   1   6]
 [  2 274   0   1   2]
 [  5   0 283   2   9]
 [  2   0   1 285   6]
 [  0   0   0   2 307]]
accuracy of NB 0.974
```

For navebays this split works between 50 and 80. Again MI is giving better value as it considers probablility of terms with other classes too in the formula while tfidf just takes frequency of that word in class and occurrence  of word in other class(idf).

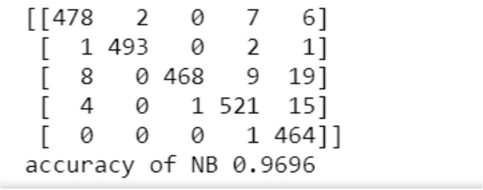
50:50

Tfidf



```
[[438   1  14   9  33]
 [  1 406   9   7  75]
 [  4   0 482   4  20]
 [  7   1   5 471  14]
 [  2   0   2   8 487]]
accuracy of NB 0.9136
```

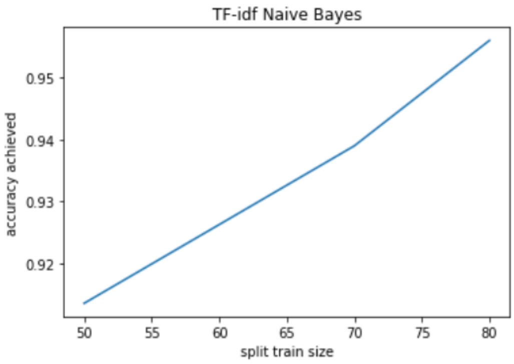
MI



```
[[478   2   0   7   6]
 [  1 493   0   2   1]
 [  8   0 468   9  19]
 [  4   0   1 521  15]
 [  0   0   0   1 464]]
accuracy of NB 0.9696
```

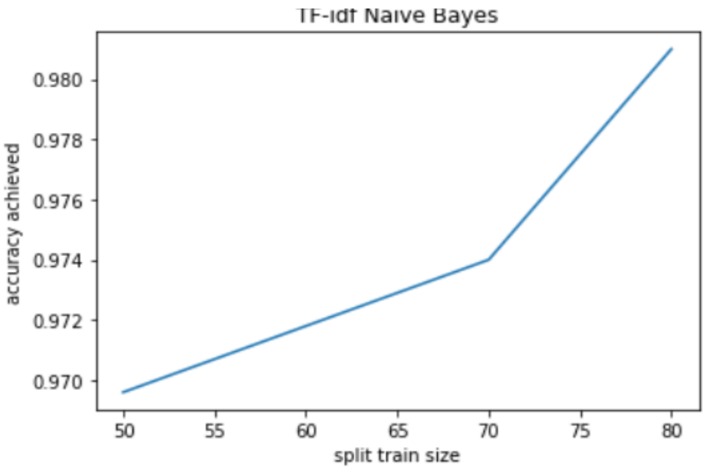
In this split performance of naive is  worst behaving still mi is better  than tfidf reason stated above.

As in mi which also see n00 ,n01,n10 .while in tfidf our main focus is just on n11. Therefore mi is performing better .



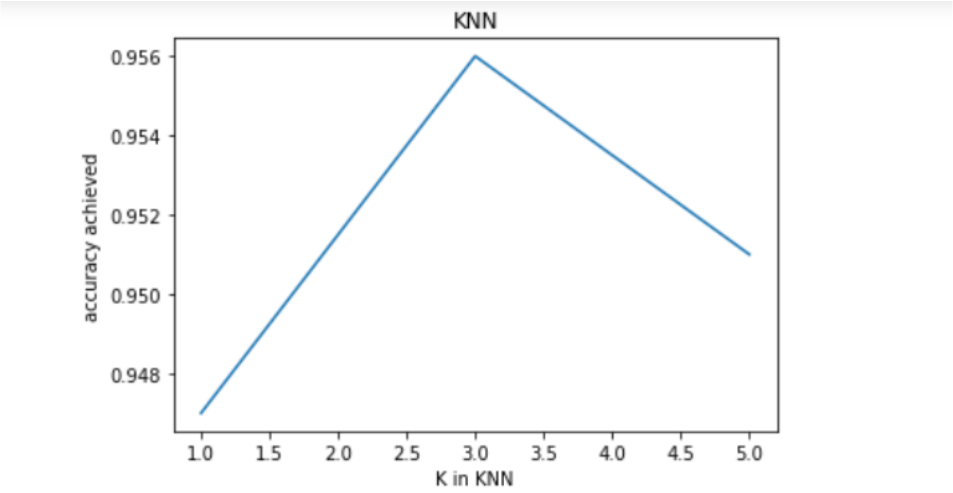
Here we can see with split train size increasing Naïve bays performs better both for tfidf and mi.

Here at train split size =80% best performance is reached.



MI Split Comparison :split 80 20 performs best

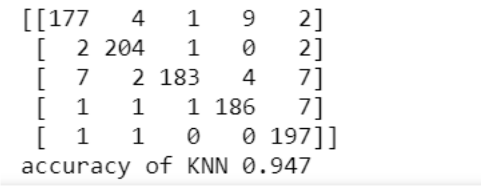
KNN  (Chosing best k for KNN)



For this plot split is taken as 80:20

At K=1

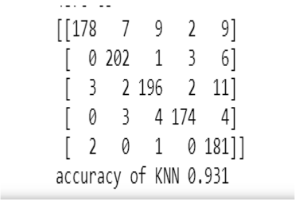
Knn performs worst at k=1 noise could  enter.as we are seeing just one nearest neighbour .



```
[[177   4   1   9   2]
 [  2 204   1   0   2]
 [  7   2 183   4   7]
 [  1   1   1 186   7]
 [  1   1   0   0 197]]
accuracy of KNN 0.947
```

At k=3

Knn perform best in this case.Usually knn performs best at odd values. It has also seen that at kequal to under root of number of classes performs best.which can be seen here. At  k =3 accuracy is highest.Therefore further experiment here k=3 is taken .



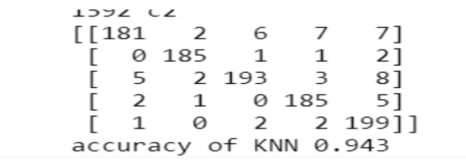
```
[[178   7   9   2   9]
 [  0 202   1   3   6]
 [  3   2 196   2  11]
 [  0   3   4 174   4]
 [  2   0   1   0 181]]
accuracy of KNN 0.931
```

Here  behaviour for mi and tfidf was same.

In Knn while testing top k  were picked and majortiy score class was given,

Here if there is one class of c1 at top 1 position but 3 docs class3 at next position then class was assigned after summing the cosine score classwise.Class which had hishest scoew was assigned.

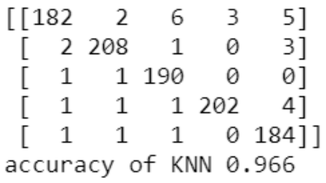
At k=5 some time accuracy was better than k=3,but it is for sure that both are better than k=1 as during outlier also it can give wrong prediction.



```
1592 c2
[[181   2   6   7   7]
 [  0 185   1   1   2]
 [  5   2 193   3   8]
 [  2   1   0 185   5]
 [  1   0   2   2 199]]
accuracy of KNN 0.943
```

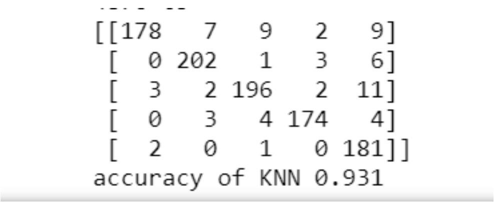
KNN  80:20 (all analysis is done for k=3 as it came out to be the best as per above analysis)

Tfidf at k=3



```
[[182   2   6   3   5]
 [  2 208   1   0   3]
 [  1   1 190   0   0]
 [  1   1   1 202   4]
 [  1   1   1   0 184]]
accuracy of KNN 0.966
```

MI



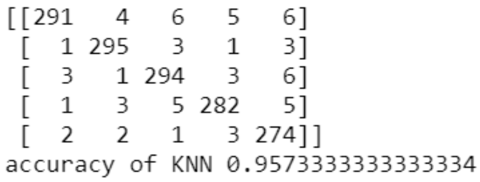
```
[[178   7   9   2   9]
 [  0 202   1   3   6]
 [  3   2 196   2  11]
 [  0   3   4 174   4]
 [  2   0   1   0 181]]
accuracy of KNN 0.931
```

These are various split accuracies for both tsidf and mi.

Here split pattern hike is not seen as there was in the naïve bays.Here 70 30 Is performing good.

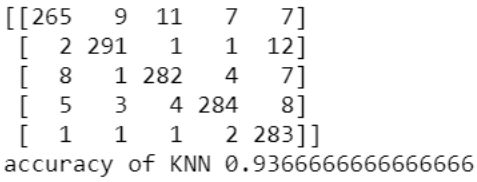
Knn at 70:30

Tfidf



```
[[291   4   6   5   6]
 [  1 295   3   1   3]
 [  3   1 294   3   6]
 [  1   3   5 282   5]
 [  2   2   1   3 274]]
accuracy of KNN 0.9573333333333334
```

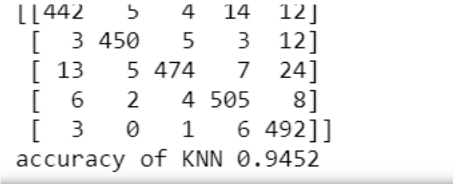
MI



```
[[265   9  11   7   7]
 [  2 291   1   1  12]
 [  8   1 282   4   7]
 [  5   3   4 284   8]
 [  1   1   1   2 283]]
accuracy of KNN 0.9366666666666666
```

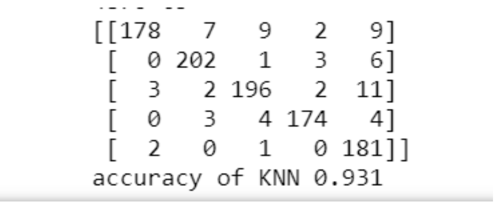
KNN at  50:50

Tfidf



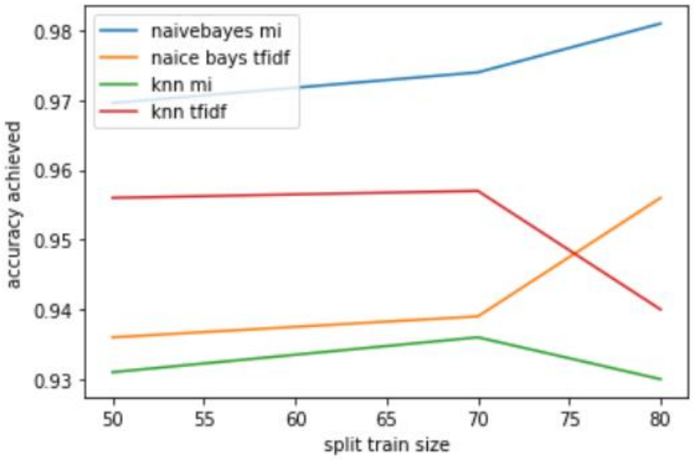
```
[[442   5   4  14  12]
 [  3 450   5   3  12]
 [ 13   5 474   7  24]
 [  6   2   4 505   8]
 [  3   0   1   6 492]]
accuracy of KNN 0.9452
```

MI



```
[[178   7   9   2   9]
 [  0 202   1   3   6]
 [  3   2 196   2  11]
 [  0   3   4 174   4]
 [  2   0   1   0 181]]
accuracy of KNN 0.931
```

Overall Comparison of all variation



Overall here Naïve bays with mutual information is performing best with assumed top k % as shown above as naïve bays takes probability into consideration and mutual information also see n00 ,n01,n10 .while in tfidf our main focus is just on n11. Therefore mi is performing better .

While in Knn tfidf is working good with k=3.Here boyth knn graphs are built as per analysis while taking k=3.