NAIVE BAYES VS. MAX ENTROPY

I HAVE MERGED AND COLLECTED MOST OF THE NEEDED READMES IN THIS PDF IF YOU CAN NOT FIND A README IN FOLDERS YOU CAN SEE THE EXPLANATION OF THAT README HERE ...

NAIVE BAYES:

The results of naive bayes can be found in the directory shown below:
NLP97982\P2\ClsModel\NaiveBayes\Test.report.txt :

Pop accuracy percentage : 93.17180616740089
Pop Precision percentage : 93.38235294117648
Pop Recall percentage : 95.13108614232209
Pop F1 percentage : 94.24860853432281

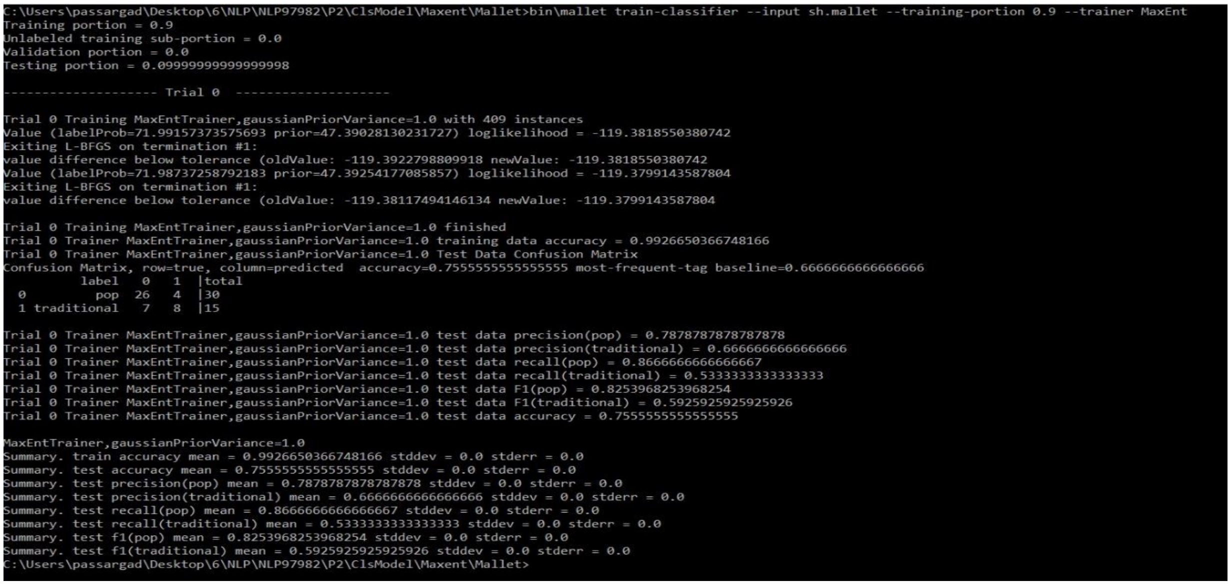
Traditional accuracy percentage : 93.17180616740089
Traditional Precision : 92.85714285714286
Traditional Recall : 90.37433155080214
Traditional F1 : 91.59891598915989

MAX ENTROPY:

1.As MaxEnt needs features first of all unigram was used as feature and a file was created in format of

--label feature:value --
that label is one of the {pop-traditional} labels
feature contains every unigram word in each sentence
value: for each word appearing in a sentence the value is 1 and if sentence includes a word more than once for example twice that word comes twice with value of 1.

By using mallet, making model, training and testing it by mallet commands the results became:

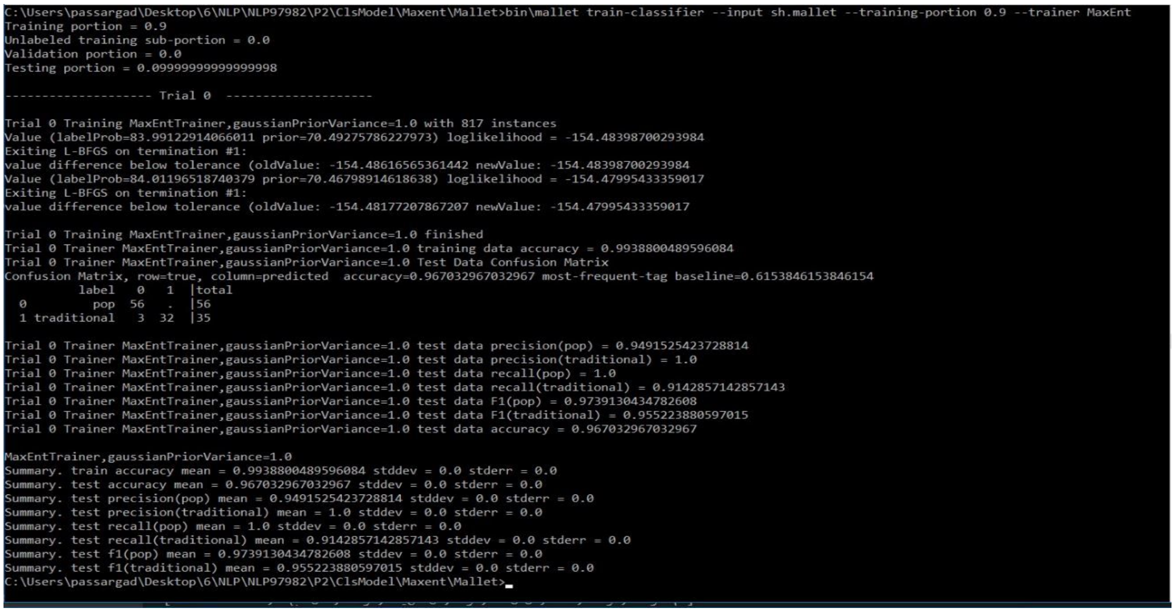


```
C:\Users\passargad\Desktop\6\NLP\NLP97982\P2\ClsModel\Maxent\Mallet>bin\mallet train-classifier --input sh.mallet --training-portion 0.9 --trainer MaxEnt
Training portion = 0.9
Unlabeled training sub-portion = 0.0
Validation portion = 0.0
Testing portion = 0.09999999999999998

-------------------- Trial 0  --------------------

Trial 0 Training MaxEntTrainer,gaussianPriorVariance=1.0 with 409 instances
Value (labelProb=71.99157373575693 prior=47.39028130231727) loglikelihood = -119.3818550380742
Exiting L-BFGS on termination #1:
value difference below tolerance (oldValue: -119.3922798809918 newValue: -119.3818550380742
Value (labelProb=71.98737258792183 prior=47.39254177085857) loglikelihood = -119.3799143587804
Exiting L-BFGS on termination #1:
value difference below tolerance (oldValue: -119.38117494146134 newValue: -119.3799143587804

Trial 0 Training MaxEntTrainer,gaussianPriorVariance=1.0 finished
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 training data accuracy = 0.9926650366748166
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 Test Data Confusion Matrix
Confusion Matrix, row=true, column=predicted  accuracy=0.7555555555555555 most-frequent-tag baseline=0.6666666666666666
          label   0   1  |total
  0         pop  26   4  |30
  1 traditional   7   8  |15

Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data precision(pop) = 0.7878787878787878
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data precision(traditional) = 0.6666666666666666
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data recall(pop) = 0.8666666666666667
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data recall(traditional) = 0.5333333333333333
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data F1(pop) = 0.8253968253968254
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data F1(traditional) = 0.5925925925925926
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data accuracy = 0.7555555555555555

MaxEntTrainer,gaussianPriorVariance=1.0
Summary. train accuracy mean = 0.9926650366748166 stddev = 0.0 stderr = 0.0
Summary. test accuracy mean = 0.7555555555555555 stddev = 0.0 stderr = 0.0
Summary. test precision(pop) mean = 0.7878787878787878 stddev = 0.0 stderr = 0.0
Summary. test precision(traditional) mean = 0.6666666666666666 stddev = 0.0 stderr = 0.0
Summary. test recall(pop) mean = 0.8666666666666667 stddev = 0.0 stderr = 0.0
Summary. test recall(traditional) mean = 0.5333333333333333 stddev = 0.0 stderr = 0.0
Summary. test f1(pop) mean = 0.8253968253968254 stddev = 0.0 stderr = 0.0
Summary. test f1(traditional) mean = 0.5925925925925926 stddev = 0.0 stderr = 0.0
C:\Users\passargad\Desktop\6\NLP\NLP97982\P2\ClsModel\Maxent\Mallet>
```

As picture shows the result was not very good and accuracy was about 75% which was not as we expected from MaxEnt.

So MaxEnt needed more features...

2.Biagram was the next choice. Mallet does not understand space between words of one features so biagram words were joined together and became as one word by following above steps the results became:

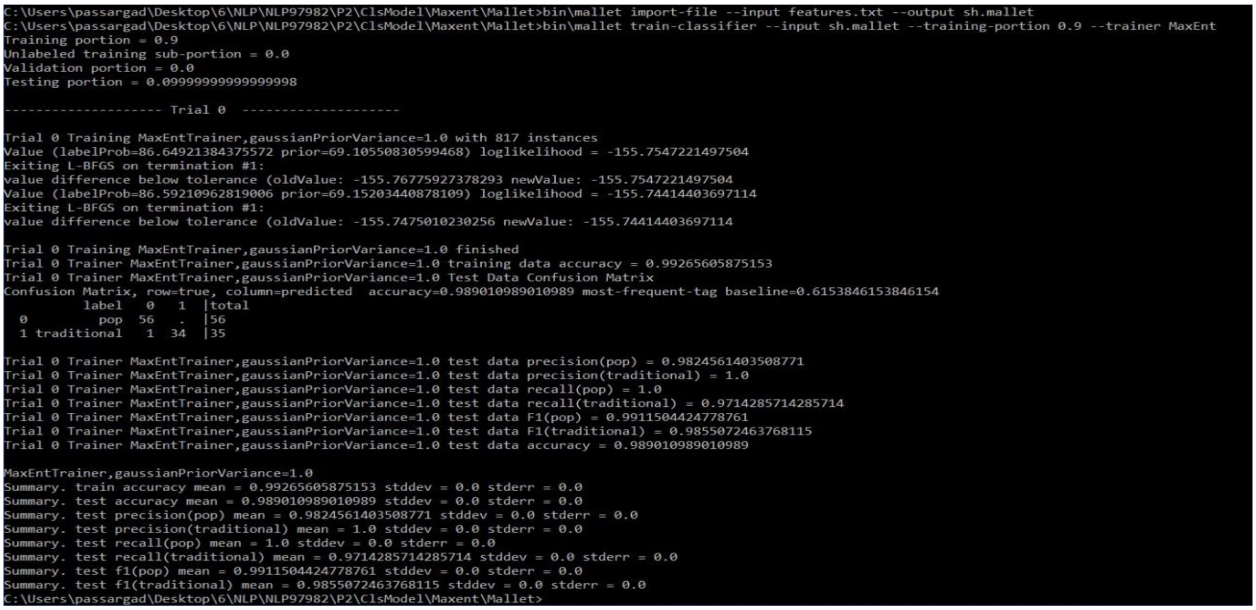


```
C:\Users\passargad\Desktop\6\NLP\NLP97982\P2\ClsModel\Maxent\Mallet>bin\mallet train-classifier --input sh.mallet --training-portion 0.9 --trainer MaxEnt
Training portion = 0.9
Unlabeled training sub-portion = 0.0
Validation portion = 0.0
Testing portion = 0.09999999999999998

-------------------- Trial 0  --------------------

Trial 0 Training MaxEntTrainer,gaussianPriorVariance=1.0 with 817 instances
Value (labelProb=83.99122914066011 prior=70.49275786227973) loglikelihood = -154.48398700293984
Exiting L-BFGS on termination #1:
value difference below tolerance (oldValue: -154.48616565361442 newValue: -154.48398700293984
Value (labelProb=84.01196518740379 prior=70.46798914618638) loglikelihood = -154.47995433359017
Exiting L-BFGS on termination #1:
value difference below tolerance (oldValue: -154.48177207867207 newValue: -154.47995433359017

Trial 0 Training MaxEntTrainer,gaussianPriorVariance=1.0 finished
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 training data accuracy = 0.9938800489596084
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 Test Data Confusion Matrix
Confusion Matrix, row=true, column=predicted  accuracy=0.967032967032967 most-frequent-tag baseline=0.6153846153846154
          label   0   1  |total
  0         pop  56   .  |56
  1 traditional   3  32  |35

Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data precision(pop) = 0.9491525423728814
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data precision(traditional) = 1.0
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data recall(pop) = 1.0
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data recall(traditional) = 0.9142857142857143
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data F1(pop) = 0.9739130434782608
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data F1(traditional) = 0.955223880597015
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data accuracy = 0.967032967032967

MaxEntTrainer,gaussianPriorVariance=1.0
Summary. train accuracy mean = 0.9938800489596084 stddev = 0.0 stderr = 0.0
Summary. test accuracy mean = 0.967032967032967 stddev = 0.0 stderr = 0.0
Summary. test precision(pop) mean = 0.9491525423728814 stddev = 0.0 stderr = 0.0
Summary. test precision(traditional) mean = 1.0 stddev = 0.0 stderr = 0.0
Summary. test recall(pop) mean = 1.0 stddev = 0.0 stderr = 0.0
Summary. test recall(traditional) mean = 0.9142857142857143 stddev = 0.0 stderr = 0.0
Summary. test f1(pop) mean = 0.9739130434782608 stddev = 0.0 stderr = 0.0
Summary. test f1(traditional) mean = 0.955223880597015 stddev = 0.0 stderr = 0.0
C:\Users\passargad\Desktop\6\NLP\NLP97982\P2\ClsModel\Maxent\Mallet>
```

Results then became much better (about 96%) but it could be better...

3.Some words give us a good view of each label for example

'آن' and 'ان' comes in traditional lyrics much more than pop lyrics.Another examples can be the words 'است' and 'هست' that they can be seen in traditional lyrics more than pop.
So they were good for becoming as MaxEnt features.
By adding them and following mallet model creater and tester commands results became:



```
C:\Users\passargad\Desktop\6\NLP\NLP97982\P2\ClsModel\Maxent\Mallet>bin\mallet import-file --input features.txt --output sh.mallet
C:\Users\passargad\Desktop\6\NLP\NLP97982\P2\ClsModel\Maxent\Mallet>bin\mallet train-classifier --input sh.mallet --training-portion 0.9 --trainer MaxEnt
Training portion = 0.9
Unlabeled training sub-portion = 0.0
Validation portion = 0.0
Testing portion = 0.09999999999999998

-------------------- Trial 0  --------------------

Trial 0 Training MaxEntTrainer,gaussianPriorVariance=1.0 with 817 instances
Value (labelProb=86.64921384375572 prior=69.10550830599468) loglikelihood = -155.7547221497504
Exiting L-BFGS on termination #1:
value difference below tolerance (oldValue: -155.76775927378293 newValue: -155.7547221497504
Value (labelProb=86.59210962819006 prior=69.15203440878109) loglikelihood = -155.74414403697114
Exiting L-BFGS on termination #1:
value difference below tolerance (oldValue: -155.7475010230256 newValue: -155.74414403697114

Trial 0 Training MaxEntTrainer,gaussianPriorVariance=1.0 finished
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 training data accuracy = 0.99265605875153
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 Test Data Confusion Matrix
Confusion Matrix, row=true, column=predicted  accuracy=0.989010989010989 most-frequent-tag baseline=0.6153846153846154
          label   0   1  |total
  0         pop  56   .  |56
  1 traditional   1  34  |35

Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data precision(pop) = 0.9824561403508771
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data precision(traditional) = 1.0
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data recall(pop) = 1.0
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data recall(traditional) = 0.9714285714285714
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data F1(pop) = 0.9911504424778761
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data F1(traditional) = 0.9855072463768115
Trial 0 Trainer MaxEntTrainer,gaussianPriorVariance=1.0 test data accuracy = 0.989010989010989

MaxEntTrainer,gaussianPriorVariance=1.0
Summary. train accuracy mean = 0.99265605875153 stddev = 0.0 stderr = 0.0
Summary. test accuracy mean = 0.989010989010989 stddev = 0.0 stderr = 0.0
Summary. test precision(pop) mean = 0.9824561403508771 stddev = 0.0 stderr = 0.0
Summary. test precision(traditional) mean = 1.0 stddev = 0.0 stderr = 0.0
Summary. test recall(pop) mean = 1.0 stddev = 0.0 stderr = 0.0
Summary. test recall(traditional) mean = 0.9714285714285714 stddev = 0.0 stderr = 0.0
Summary. test f1(pop) mean = 0.9911504424778761 stddev = 0.0 stderr = 0.0
Summary. test f1(traditional) mean = 0.9855072463768115 stddev = 0.0 stderr = 0.0
C:\Users\passargad\Desktop\6\NLP\NLP97982\P2\ClsModel\Maxent\Mallet>
```

THE ACCURACY IS 98% NOW AND IT'S A GOOD ADVANCE IN ADDING USABLE FEATURES ^__^
Of course there can be added much more features too...

The final accuracy of MaxEnt is better than naive bayes and means that if we have good features max entropy helps us better but in the case which we can not find special features to distinguish our labels naive bayes can be more helpful.

THE END