CS 529:
Introduction to Machine Learning

Project 2
on
Naive Bayes Classifier

Submitted by: Veera Venakata Sathya Bhargav Nunna

Question 1:
Given a document, the 1000 words the document contains may be arranged in any order. So the position of a particular word is not fixated to the observed position. When the document is changed in the sequence of words without altering the meaning, the first observed word may now appear at a different position. Thus it would be difficult to comprehend the paramters for 1000 such words from the 50k word bank.

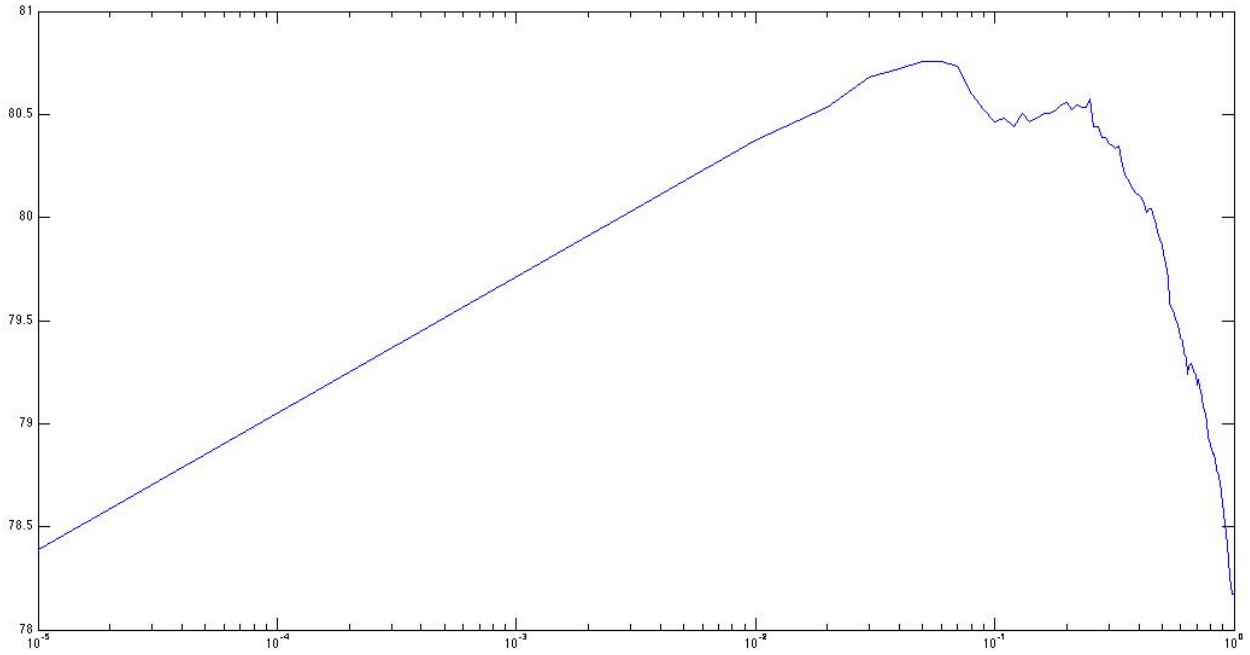
Question 2:
Accuracy 78.5 % .

The confusion matrix is

249	0	0	0	0	1	0	0	1	0	0	2	0	3	3	24	2	3	4	26
0	286	13	14	9	22	4	1	1	0	1	11	8	6	10	1	2	0	0	0
1	33	204	57	19	21	4	2	3	0	0	12	5	10	8	3	1	0	5	3
0	11	30	277	20	1	10	2	1	0	1	4	32	1	2	0	0	0	0	0
0	17	13	30	269	0	12	2	2	0	0	3	21	8	4	0	1	0	1	0
0	54	16	6	3	285	1	1	3	0	0	5	3	6	4	0	1	1	1	0
0	7	5	32	16	1	270	17	8	1	2	0	7	4	6	0	2	1	2	1
0	3	1	2	0	0	14	331	17	0	0	1	13	0	4	2	0	0	6	1
0	1	0	1	0	0	2	27	360	0	0	0	3	1	0	0	1	1	0	0
0	0	0	1	1	0	2	1	2	352	17	0	1	3	3	5	2	1	5	1
2	0	1	0	0	0	2	1	2	4	383	0	0	0	0	1	2	0	1	0
0	3	0	3	4	1	0	0	0	1	1	362	2	2	2	0	9	0	5	0
3	20	4	25	7	4	8	11	6	0	0	21	264	9	7	1	3	0	0	0
5	7	0	3	0	0	3	5	4	1	0	1	8	320	8	7	6	5	8	2
0	8	0	1	0	3	1	0	1	0	1	4	6	5	343	3	2	1	12	1
11	2	0	0	0	2	1	0	0	0	0	0	0	2	0	362	0	1	2	15
1	1	0	0	0	1	1	2	1	1	0	4	0	5	2	1	303	5	23	13
12	1	0	1	0	0	1	2	0	2	0	2	1	0	0	6	3	326	18	1
6	1	0	0	1	1	0	0	0	0	0	5	0	10	6	2	63	6	196	13
39	3	0	0	0	0	0	0	1	1	0	1	0	2	6	27	10	3	7	151

Question 3: *Yes, there are newsgroups that may fail the algorithm in classifying accurately because of the words that were used in the document. All the documents taken from a single news group, presenting news about one of the topics would definitely use some words that would be common to all the documents. In such cases the algorithm may misclassify the words as it is confused.*

Question 4:



Question 5:

I propose the method of Information gain
As we are using a Navies Bayes classifier, a Navies Bayes is given by
$P(A|B)= P(B|A)P(A)$
Relating this with our labels and words
$P(L_j|W_i) = P(W_i|L_j)P(L_j)$
=>Document frequency = Term frequency prior*

Document frequency is the probability of a word 'i' occurring in Label j. The value of this probability states whether the word occurs in too frequently in documents or is it a special word. If the value is low, then the word is unique and helps in better classification, which is what we need here. While, term frequency defines the probability of a word occurring in a given label. This frequency ought to be higher so that it shows that the word has a high probability of occurring in the label.

Entropy can be written in terms of these probabilities.

$E(S) = -P(L_j|W_i)\log P(L_j|W_i) -(1- P(L_j|W_i))\log(1- P(L_j|W_i))$

And $E(S|X) = - P(W_i|L_j)\log P(W_i|L_j) - (1- P(W_i|L_j))\log(1- P(W_i|L_j))$

Thus we can calculate the information gain from the above formulae. But it is known that Information gain is approx. equal to Term frequency id. But the rank of the word is proportional to Term frequency id. Therefore, the information gain is proportional to rank of the word, which is the final result we want.

Question 6:

My top hundred words are as follows

['windows' 'god' 'he' 'scsi' 'car' 'drive' 'space' 'team' 'dos' 'bike' 'file' 'of' 'that' 'mb' 'game' 'key' 'mac' 'jesus' 'window' 'dod' 'hockey' 'the' 'graphics' 'card' 'image' 'his' 'gun' 'encryption' 'sale' 'apple' 'government' 'season' 'we' 'games' 'israel' 'disk' 'files' 'ide' 'controller' 'players' 'shipping' 'chip' 'program' 'was' 'cars' 'nasa' 'win' 'year' 'were' 'they' 'turkish' 'motif' 'people' 'armenian' 'play' 'drives' 'bible' 'use' 'widget' 'pc' 'clipper' 'offer' 'jpeg' 'baseball' 'bus' 'my' 'nhl' 'software' 'is' 'db' 'server' 'jews' 'os' 'israeli' 'output' 'data' 'system' 'who' 'league' 'armenians' 'for' 'christian' 'christians' 'entry' 'mhz' 'ftp' 'price' 'christ' 'guns' 'thanks' 'church' 'color' 'teams' 'privacy' 'condition' 'launch' 'him' 'com' 'monitor' 'ram']

Question 7:

There are some signs of the given dataset to be biased. I could say so because, when I went through the dataset I found some words such as 'aarp', 'dingell', 'qkcok' and many more that have a very rare chance of being included in the future real world testing data, according to me. So as the classifier is trained so much on the present training data, that it may form the condition of 'overfitting' like in decision trees. So when the classifier is applied on testing data, which was trained on this data, there is a high probability that the given results would not be accurate. Therefore there is a chance that this training data may not help the classifier in learning well and performing well in the future. Thus I conclude that the data is biased.