GEROSH SHIBU GEORGE

19BCE1403

WM LAB - 9

Naive Bayes algorithm lab assignment

Steps:

1. An excel file is provided which contains training data and test data.
2. Find out all the unique words in **training and test data** to find all the vocabulary contents. Find the size of vocabulary |V| first.
3. Find the number of documents belonging to an individual class in training data alone. Divide by total number of documents to get prior probability of that class.

$$\hat{P}(c) = \frac{N_c}{N}$$

4. If you add all prior class probabilities it will come to 1.
5. Now form a dictionary for ever class containing all the words and their frequency.
6. Count the total number of words for every class.

$$\hat{P}(w \mid c) = \frac{count(w,c)+1}{count(c)+|V|}$$

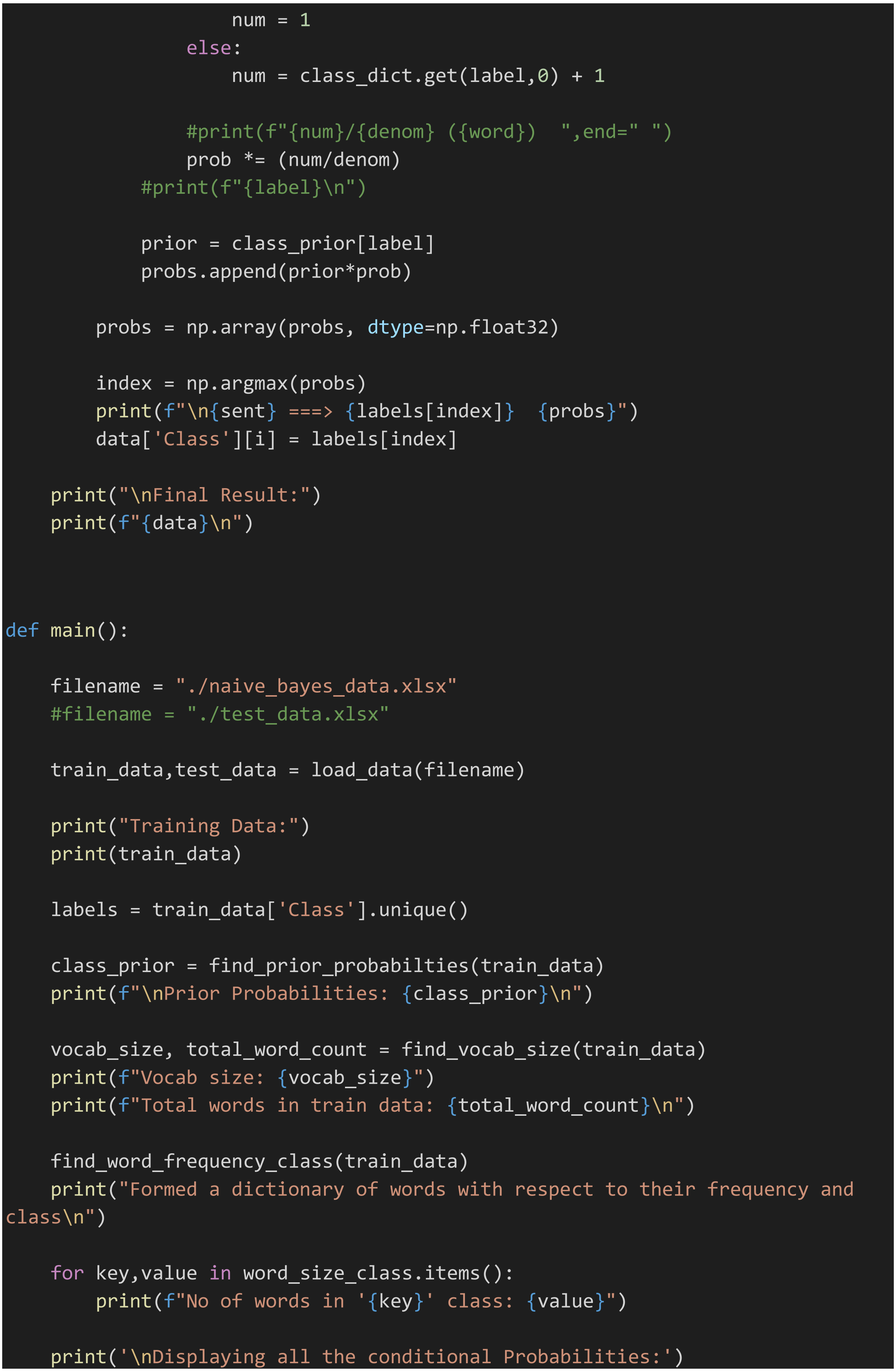
7. use the above formula to calculate the conditional probabilities of a word belonging to a particular class. [calculate for all the words happening in a chosen test sentence.]
8. Now determine the probability of the chosen test document belonging to a class

$$\hat{P}(c)\prod_{i}\hat{P}(x_i \mid c)$$

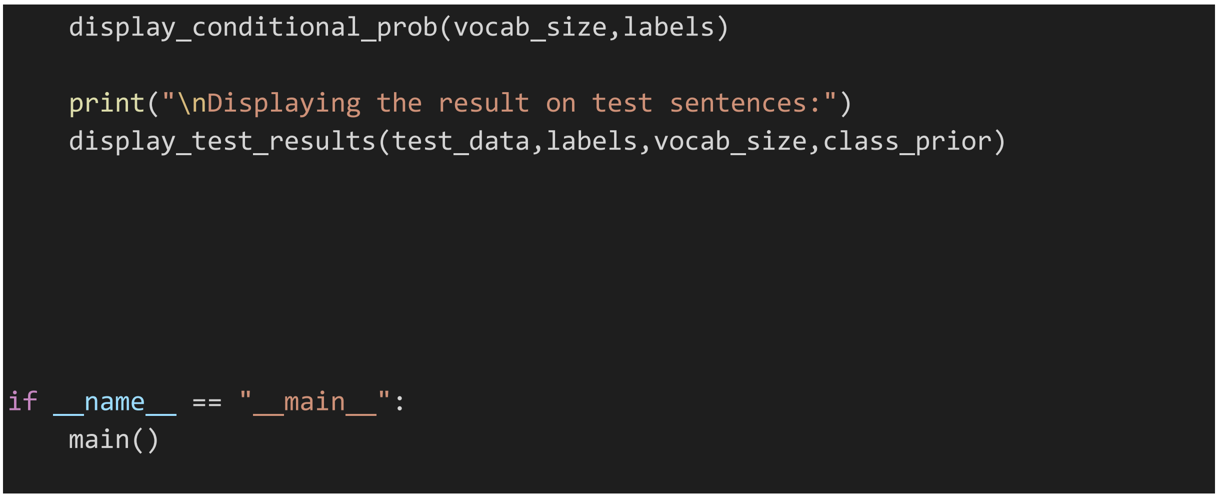
9. Find the maximum conditional probability and identify that class. Print out that sentence and the name of the class.
   Repeat for all sentences in the test data [steps 7-9].

CODE:

```
import pandas as pd
import numpy as np
import re

all_words = []
word_size_class = {}
all_sents = []
word_frequency_label = {}

def load_data(filename):

    data = pd.read_excel(filename)
    test_index = data['Class'] == '?'
    training_data = data[-test_index]
    testing_data = data[test_index]
    testing_data.reset_index(inplace = True, drop = True)

    return (training_data,testing_data)

def find_vocab_size(training_data):

    for index,sent in enumerate(training_data['URL']):
        ext_words = re.findall(r"([a-z0-9]+)",sent)
        label = training_data['Class'][index]
        word_size_class[label] = word_size_class.get(label,0) + len(ext_words)
        all_words.extend(ext_words)
        all_sents.append(ext_words)

    unique_words_count = len(set(all_words))
    all_words_count = len(all_words)

    global unique_words
    unique_words = list(set(all_words))

    return (unique_words_count,all_words_count)


def find_prior_probabilties(training_data):

    class_prior = {}

    labels = training_data['Class'].unique()

    total = len(training_data)
    for l in labels:
        class_prior[l] = sum(training_data['Class'] == l) / total
    return class_prior


def find_word_frequency_class(training_data):

    for word in unique_words:

        for index,sent_vec in enumerate(all_sents):
            if word in sent_vec:

                if word not in word_frequency_label:
                    word_frequency_label[word] = {}
                label = training_data['Class'][index]

                word_frequency_label[word][label] = word_frequency_label[word].get(label,0) + sent_vec.count(word)


def display_conditional_prob(vocab_size,labels):

    i=0
    for word in word_frequency_label:

        for label in labels:
            num = word_frequency_label[word].get(label,0) + 1
            denom = word_size_class[label] + vocab_size
            space = " "
            print(f"P({word}/{label}) = {num}/{denom} {space*(14-len(word))}",end="\t")

        print()

def display_test_results(data,labels,vocab_size,class_prior):


    for i,sent in enumerate(data['URL']):
        ext_words = re.findall(r"([a-z0-9]+)",sent)

        probs = []
        for label in labels:
            prob = 1
            for word in ext_words:

                class_dict = word_frequency_label.get(word)

                num = 0
                denom = (word_size_class[label]+vocab_size)

                if class_dict == None:
```

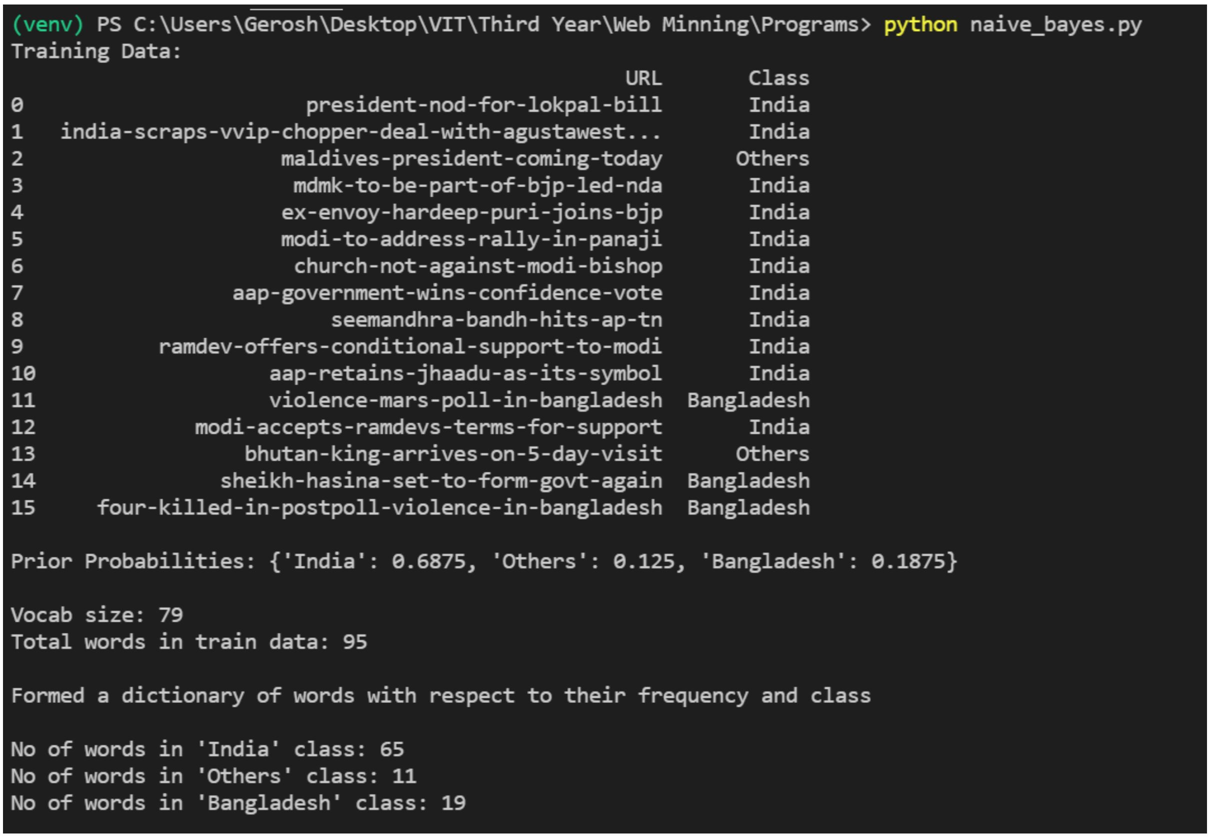


```
                    num = 1
                else:
                    num = class_dict.get(label,0) + 1

                #print(f"{num}/{denom} ({word})  ",end=" ")
                prob *= (num/denom)
            #print(f"{label}\n")

            prior = class_prior[label]
            probs.append(prior*prob)

        probs = np.array(probs, dtype=np.float32)

        index = np.argmax(probs)
        print(f"\n{sent} ===> {labels[index]}  {probs}")
        data['Class'][i] = labels[index]

    print("\nFinal Result:")
    print(f"{data}\n")



def main():

    filename = "./naive_bayes_data.xlsx"
    #filename = "./test_data.xlsx"

    train_data,test_data = load_data(filename)

    print("Training Data:")
    print(train_data)

    labels = train_data['Class'].unique()

    class_prior = find_prior_probabilties(train_data)
    print(f"\nPrior Probabilities: {class_prior}\n")

    vocab_size, total_word_count = find_vocab_size(train_data)
    print(f"Vocab size: {vocab_size}")
    print(f"Total words in train data: {total_word_count}\n")

    find_word_frequency_class(train_data)
    print("Formed a dictionary of words with respect to their frequency and
class\n")

    for key,value in word_size_class.items():
        print(f"No of words in '{key}' class: {value}")

    print('\nDisplaying all the conditional Probabilities:')
```

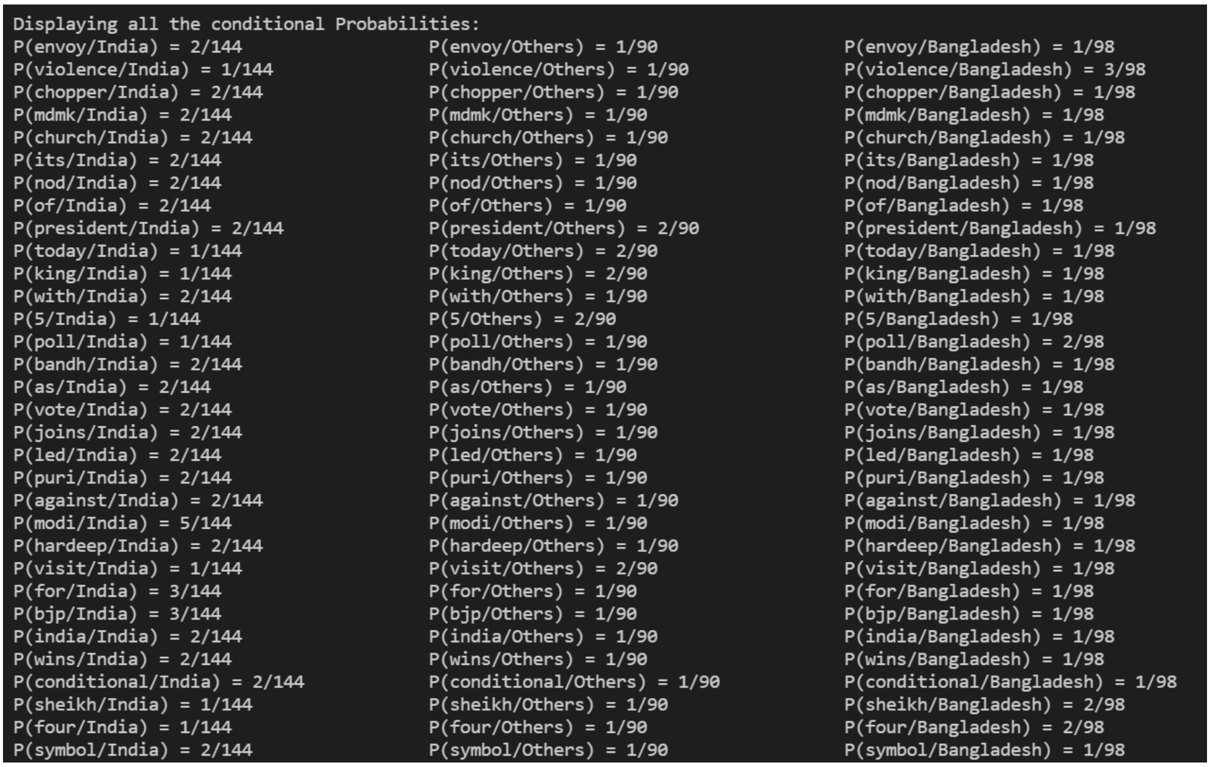


```
    display_conditional_prob(vocab_size,labels)

    print("\nDisplaying the result on test sentences:")
    display_test_results(test_data,labels,vocab_size,class_prior)





if __name__ == "__main__":
    main()
```

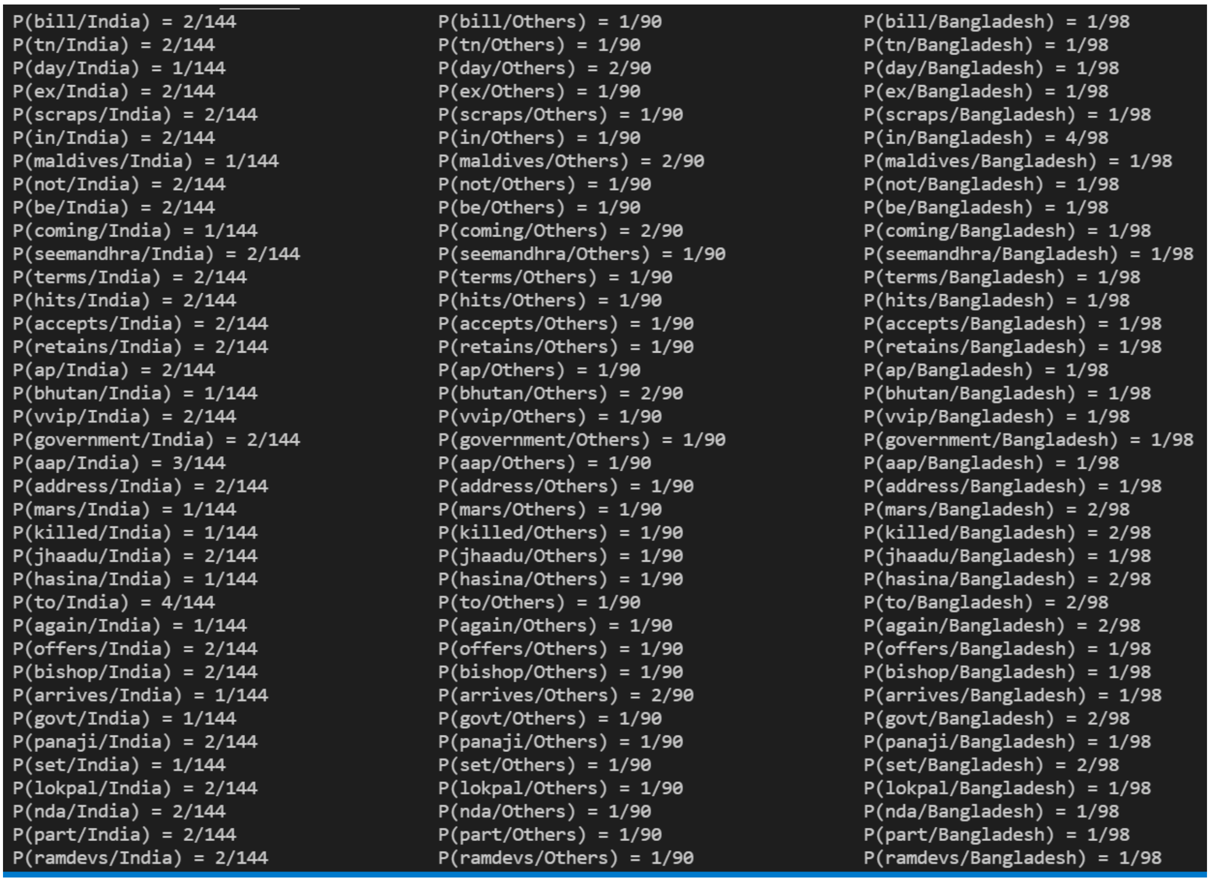
OUTPUT:



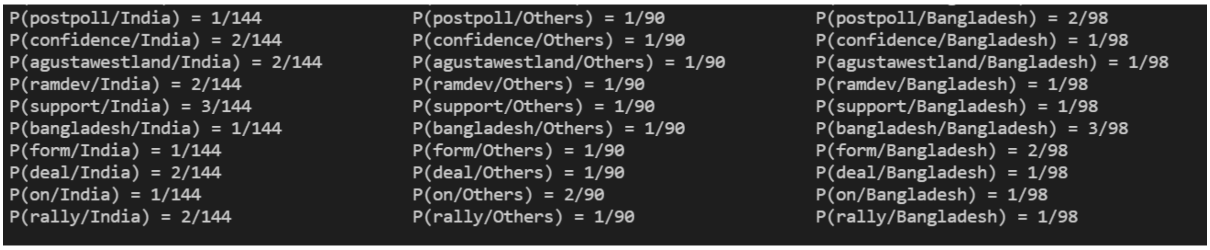
```
(venv) PS C:\Users\Gerosh\Desktop\VIT\Third Year\Web Minning\Programs> python naive_bayes.py
Training Data:
                                                  URL       Class
0                        president-nod-for-lokpal-bill       India
1    india-scraps-vvip-chopper-deal-with-agustawest...       India
2                      maldives-president-coming-today      Others
3                       mdmk-to-be-part-of-bjp-led-nda       India
4                      ex-envoy-hardeep-puri-joins-bjp       India
5                      modi-to-address-rally-in-panaji       India
6                       church-not-against-modi-bishop       India
7                  aap-government-wins-confidence-vote       India
8                          seemandhra-bandh-hits-ap-tn       India
9            ramdev-offers-conditional-support-to-modi       India
10                    aap-retains-jhaadu-as-its-symbol       India
11                    violence-mars-poll-in-bangladesh  Bangladesh
12               modi-accepts-ramdevs-terms-for-support       India
13                   bhutan-king-arrives-on-5-day-visit      Others
14                 sheikh-hasina-set-to-form-govt-again  Bangladesh
15      four-killed-in-postpoll-violence-in-bangladesh  Bangladesh

Prior Probabilities: {'India': 0.6875, 'Others': 0.125, 'Bangladesh': 0.1875}

Vocab size: 79
Total words in train data: 95

Formed a dictionary of words with respect to their frequency and class

No of words in 'India' class: 65
No of words in 'Others' class: 11
No of words in 'Bangladesh' class: 19
```



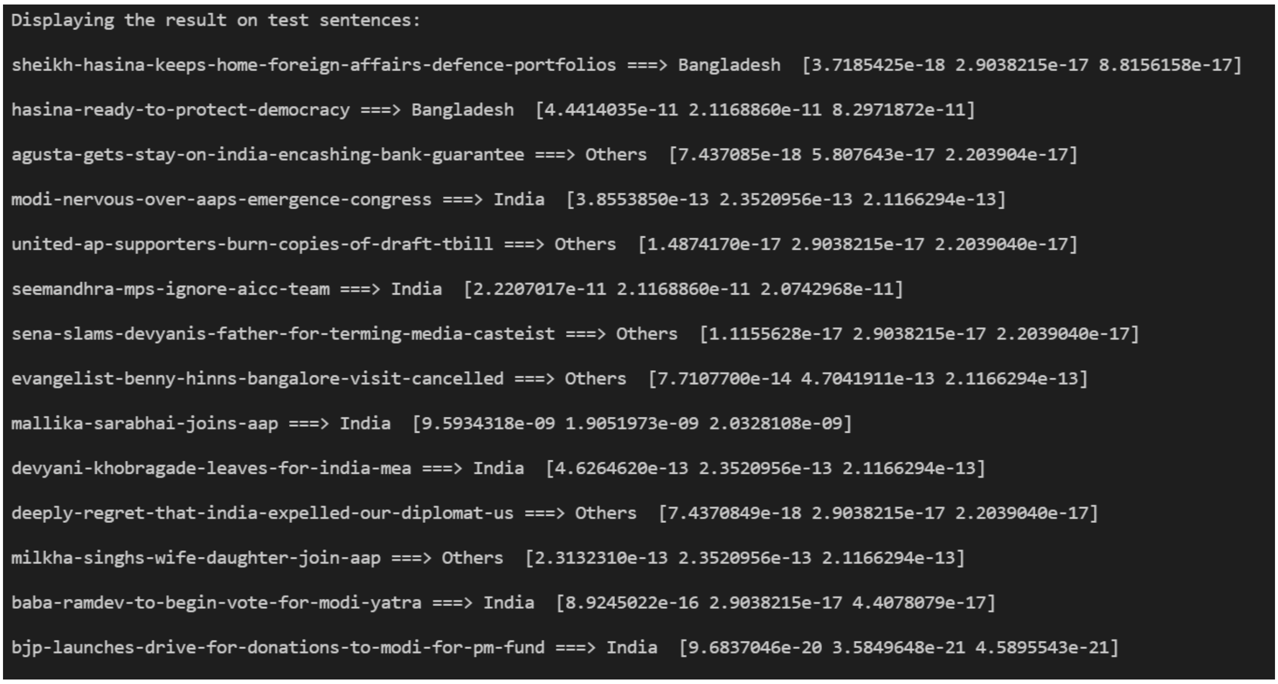
```
Displaying all the conditional Probabilities:
P(envoy/India) = 2/144                  P(envoy/Others) = 1/90                  P(envoy/Bangladesh) = 1/98
P(violence/India) = 1/144               P(violence/Others) = 1/90               P(violence/Bangladesh) = 3/98
P(chopper/India) = 2/144                P(chopper/Others) = 1/90                P(chopper/Bangladesh) = 1/98
P(mdmk/India) = 2/144                   P(mdmk/Others) = 1/90                   P(mdmk/Bangladesh) = 1/98
P(church/India) = 2/144                 P(church/Others) = 1/90                 P(church/Bangladesh) = 1/98
P(its/India) = 2/144                    P(its/Others) = 1/90                    P(its/Bangladesh) = 1/98
P(nod/India) = 2/144                    P(nod/Others) = 1/90                    P(nod/Bangladesh) = 1/98
P(of/India) = 2/144                     P(of/Others) = 1/90                     P(of/Bangladesh) = 1/98
P(president/India) = 2/144              P(president/Others) = 2/90              P(president/Bangladesh) = 1/98
P(today/India) = 1/144                  P(today/Others) = 2/90                  P(today/Bangladesh) = 1/98
P(king/India) = 1/144                   P(king/Others) = 2/90                   P(king/Bangladesh) = 1/98
P(with/India) = 2/144                   P(with/Others) = 1/90                   P(with/Bangladesh) = 1/98
P(5/India) = 1/144                      P(5/Others) = 2/90                      P(5/Bangladesh) = 1/98
P(poll/India) = 1/144                   P(poll/Others) = 1/90                   P(poll/Bangladesh) = 2/98
P(bandh/India) = 2/144                  P(bandh/Others) = 1/90                  P(bandh/Bangladesh) = 1/98
P(as/India) = 2/144                     P(as/Others) = 1/90                     P(as/Bangladesh) = 1/98
P(vote/India) = 2/144                   P(vote/Others) = 1/90                   P(vote/Bangladesh) = 1/98
P(joins/India) = 2/144                  P(joins/Others) = 1/90                  P(joins/Bangladesh) = 1/98
P(led/India) = 2/144                    P(led/Others) = 1/90                    P(led/Bangladesh) = 1/98
P(puri/India) = 2/144                   P(puri/Others) = 1/90                   P(puri/Bangladesh) = 1/98
P(against/India) = 2/144                P(against/Others) = 1/90                P(against/Bangladesh) = 1/98
P(modi/India) = 5/144                   P(modi/Others) = 1/90                   P(modi/Bangladesh) = 1/98
P(hardeep/India) = 2/144                P(hardeep/Others) = 1/90                P(hardeep/Bangladesh) = 1/98
P(visit/India) = 1/144                  P(visit/Others) = 2/90                  P(visit/Bangladesh) = 1/98
P(for/India) = 3/144                    P(for/Others) = 1/90                    P(for/Bangladesh) = 1/98
P(bjp/India) = 3/144                    P(bjp/Others) = 1/90                    P(bjp/Bangladesh) = 1/98
P(india/India) = 2/144                  P(india/Others) = 1/90                  P(india/Bangladesh) = 1/98
P(wins/India) = 2/144                   P(wins/Others) = 1/90                   P(wins/Bangladesh) = 1/98
P(conditional/India) = 2/144            P(conditional/Others) = 1/90            P(conditional/Bangladesh) = 1/98
P(sheikh/India) = 1/144                 P(sheikh/Others) = 1/90                 P(sheikh/Bangladesh) = 2/98
P(four/India) = 1/144                   P(four/Others) = 1/90                   P(four/Bangladesh) = 2/98
P(symbol/India) = 2/144                 P(symbol/Others) = 1/90                 P(symbol/Bangladesh) = 1/98
```



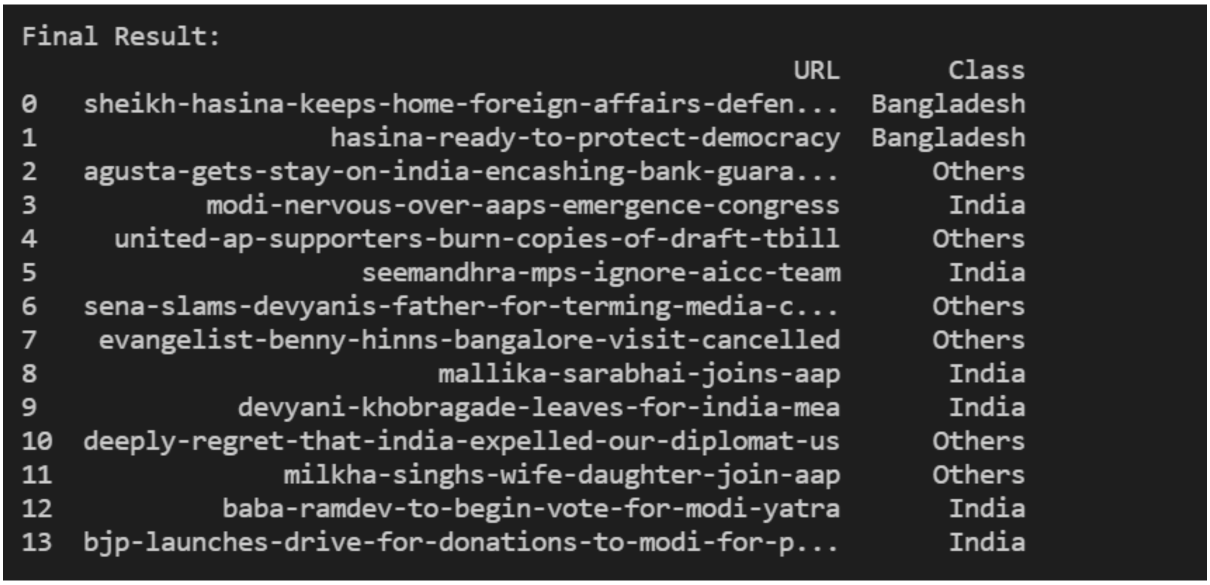
```
P(bill/India) = 2/144                    P(bill/Others) = 1/90                    P(bill/Bangladesh) = 1/98
P(tn/India) = 2/144                      P(tn/Others) = 1/90                      P(tn/Bangladesh) = 1/98
P(day/India) = 1/144                     P(day/Others) = 2/90                     P(day/Bangladesh) = 1/98
P(ex/India) = 2/144                      P(ex/Others) = 1/90                      P(ex/Bangladesh) = 1/98
P(scraps/India) = 2/144                  P(scraps/Others) = 1/90                  P(scraps/Bangladesh) = 1/98
P(in/India) = 2/144                      P(in/Others) = 1/90                      P(in/Bangladesh) = 4/98
P(maldives/India) = 1/144                P(maldives/Others) = 2/90                P(maldives/Bangladesh) = 1/98
P(not/India) = 2/144                     P(not/Others) = 1/90                     P(not/Bangladesh) = 1/98
P(be/India) = 2/144                      P(be/Others) = 1/90                      P(be/Bangladesh) = 1/98
P(coming/India) = 1/144                  P(coming/Others) = 2/90                  P(coming/Bangladesh) = 1/98
P(seemandhra/India) = 2/144              P(seemandhra/Others) = 1/90              P(seemandhra/Bangladesh) = 1/98
P(terms/India) = 2/144                   P(terms/Others) = 1/90                   P(terms/Bangladesh) = 1/98
P(hits/India) = 2/144                    P(hits/Others) = 1/90                    P(hits/Bangladesh) = 1/98
P(accepts/India) = 2/144                 P(accepts/Others) = 1/90                 P(accepts/Bangladesh) = 1/98
P(retains/India) = 2/144                 P(retains/Others) = 1/90                 P(retains/Bangladesh) = 1/98
P(ap/India) = 2/144                      P(ap/Others) = 1/90                      P(ap/Bangladesh) = 1/98
P(bhutan/India) = 1/144                  P(bhutan/Others) = 2/90                  P(bhutan/Bangladesh) = 1/98
P(vvip/India) = 2/144                    P(vvip/Others) = 1/90                    P(vvip/Bangladesh) = 1/98
P(government/India) = 2/144              P(government/Others) = 1/90              P(government/Bangladesh) = 1/98
P(aap/India) = 3/144                     P(aap/Others) = 1/90                     P(aap/Bangladesh) = 1/98
P(address/India) = 2/144                 P(address/Others) = 1/90                 P(address/Bangladesh) = 1/98
P(mars/India) = 1/144                    P(mars/Others) = 1/90                    P(mars/Bangladesh) = 2/98
P(killed/India) = 1/144                  P(killed/Others) = 1/90                  P(killed/Bangladesh) = 2/98
P(jhaadu/India) = 2/144                  P(jhaadu/Others) = 1/90                  P(jhaadu/Bangladesh) = 1/98
P(hasina/India) = 1/144                  P(hasina/Others) = 1/90                  P(hasina/Bangladesh) = 2/98
P(to/India) = 4/144                      P(to/Others) = 1/90                      P(to/Bangladesh) = 2/98
P(again/India) = 1/144                   P(again/Others) = 1/90                   P(again/Bangladesh) = 2/98
P(offers/India) = 2/144                  P(offers/Others) = 1/90                  P(offers/Bangladesh) = 1/98
P(bishop/India) = 2/144                  P(bishop/Others) = 1/90                  P(bishop/Bangladesh) = 1/98
P(arrives/India) = 1/144                 P(arrives/Others) = 2/90                 P(arrives/Bangladesh) = 1/98
P(govt/India) = 1/144                    P(govt/Others) = 1/90                    P(govt/Bangladesh) = 2/98
P(panaji/India) = 2/144                  P(panaji/Others) = 1/90                  P(panaji/Bangladesh) = 1/98
P(set/India) = 1/144                     P(set/Others) = 1/90                     P(set/Bangladesh) = 2/98
P(lokpal/India) = 2/144                  P(lokpal/Others) = 1/90                  P(lokpal/Bangladesh) = 1/98
P(nda/India) = 2/144                     P(nda/Others) = 1/90                     P(nda/Bangladesh) = 1/98
P(part/India) = 2/144                    P(part/Others) = 1/90                    P(part/Bangladesh) = 1/98
P(ramdevs/India) = 2/144                 P(ramdevs/Others) = 1/90                 P(ramdevs/Bangladesh) = 1/98
```



```
P(postpoll/India) = 1/144            P(postpoll/Others) = 1/90            P(postpoll/Bangladesh) = 2/98
P(confidence/India) = 2/144          P(confidence/Others) = 1/90          P(confidence/Bangladesh) = 1/98
P(agustawestland/India) = 2/144      P(agustawestland/Others) = 1/90      P(agustawestland/Bangladesh) = 1/98
P(ramdev/India) = 2/144              P(ramdev/Others) = 1/90              P(ramdev/Bangladesh) = 1/98
P(support/India) = 3/144             P(support/Others) = 1/90             P(support/Bangladesh) = 1/98
P(bangladesh/India) = 1/144          P(bangladesh/Others) = 1/90          P(bangladesh/Bangladesh) = 3/98
P(form/India) = 1/144                P(form/Others) = 1/90                P(form/Bangladesh) = 2/98
P(deal/India) = 2/144                P(deal/Others) = 1/90                P(deal/Bangladesh) = 1/98
P(on/India) = 1/144                  P(on/Others) = 2/90                  P(on/Bangladesh) = 1/98
P(rally/India) = 2/144               P(rally/Others) = 1/90               P(rally/Bangladesh) = 1/98
```



```
Displaying the result on test sentences:

sheikh-hasina-keeps-home-foreign-affairs-defence-portfolios ===> Bangladesh  [3.7185425e-18 2.9038215e-17 8.8156158e-17]

hasina-ready-to-protect-democracy ===> Bangladesh  [4.4414035e-11 2.1168860e-11 8.2971872e-11]

agusta-gets-stay-on-india-encashing-bank-guarantee ===> Others  [7.437085e-18 5.807643e-17 2.203904e-17]

modi-nervous-over-aaps-emergence-congress ===> India  [3.8553850e-13 2.3520956e-13 2.1166294e-13]

united-ap-supporters-burn-copies-of-draft-tbill ===> Others  [1.4874170e-17 2.9038215e-17 2.2039040e-17]

seemandhra-mps-ignore-aicc-team ===> India  [2.2207017e-11 2.1168860e-11 2.0742968e-11]

sena-slams-devyanis-father-for-terming-media-casteist ===> Others  [1.1155628e-17 2.9038215e-17 2.2039040e-17]

evangelist-benny-hinns-bangalore-visit-cancelled ===> Others  [7.7107700e-14 4.7041911e-13 2.1166294e-13]

mallika-sarabhai-joins-aap ===> India  [9.5934318e-09 1.9051973e-09 2.0328108e-09]

devyani-khobragade-leaves-for-india-mea ===> India  [4.6264620e-13 2.3520956e-13 2.1166294e-13]

deeply-regret-that-india-expelled-our-diplomat-us ===> Others  [7.4370849e-18 2.9038215e-17 2.2039040e-17]

milkha-singhs-wife-daughter-join-aap ===> Others  [2.3132310e-13 2.3520956e-13 2.1166294e-13]

baba-ramdev-to-begin-vote-for-modi-yatra ===> India  [8.9245022e-16 2.9038215e-17 4.4078079e-17]

bjp-launches-drive-for-donations-to-modi-for-pm-fund ===> India  [9.6837046e-20 3.5849648e-21 4.5895543e-21]
```



```
Final Result:
                                                  URL       Class
0   sheikh-hasina-keeps-home-foreign-affairs-defen...  Bangladesh
1                   hasina-ready-to-protect-democracy  Bangladesh
2   agusta-gets-stay-on-india-encashing-bank-guara...      Others
3           modi-nervous-over-aaps-emergence-congress       India
4     united-ap-supporters-burn-copies-of-draft-tbill      Others
5                     seemandhra-mps-ignore-aicc-team       India
6   sena-slams-devyanis-father-for-terming-media-c...      Others
7    evangelist-benny-hinns-bangalore-visit-cancelled      Others
8                          mallika-sarabhai-joins-aap       India
9             devyani-khobragade-leaves-for-india-mea       India
10  deeply-regret-that-india-expelled-our-diplomat-us      Others
11               milkha-singhs-wife-daughter-join-aap      Others
12           baba-ramdev-to-begin-vote-for-modi-yatra       India
13  bjp-launches-drive-for-donations-to-modi-for-p...       India
```