WEB MINING LAB

Faculty: Dr.Sridhar.R

LAB10
DATE: 18TH NOV 2021

VIT CHENNAI

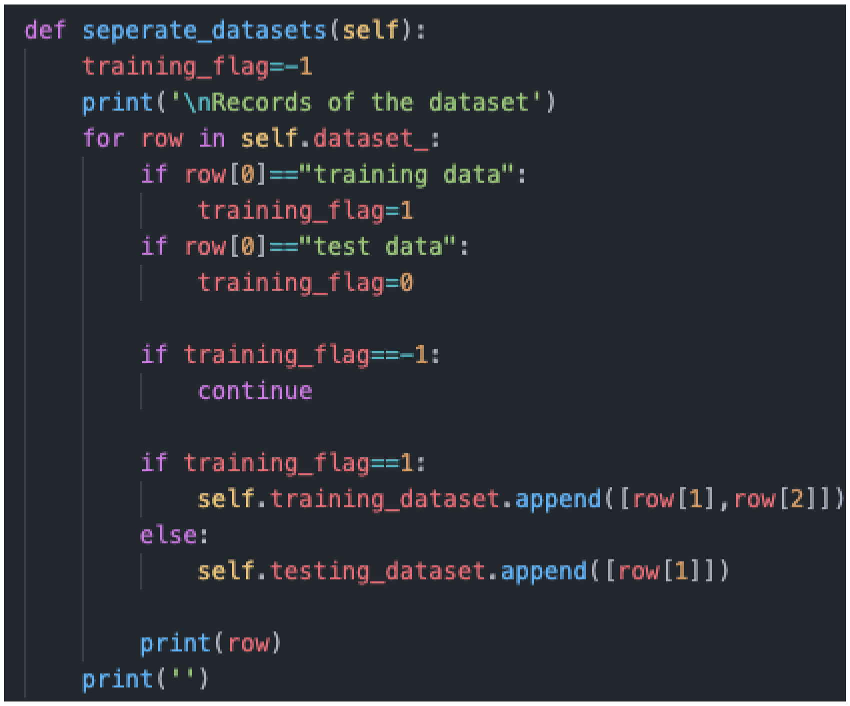
BELIDA KARTHIK
19BCE1446

Aim: To implement and understand naive Bayes classified

1. An excel file is provided which contains training data and test data.

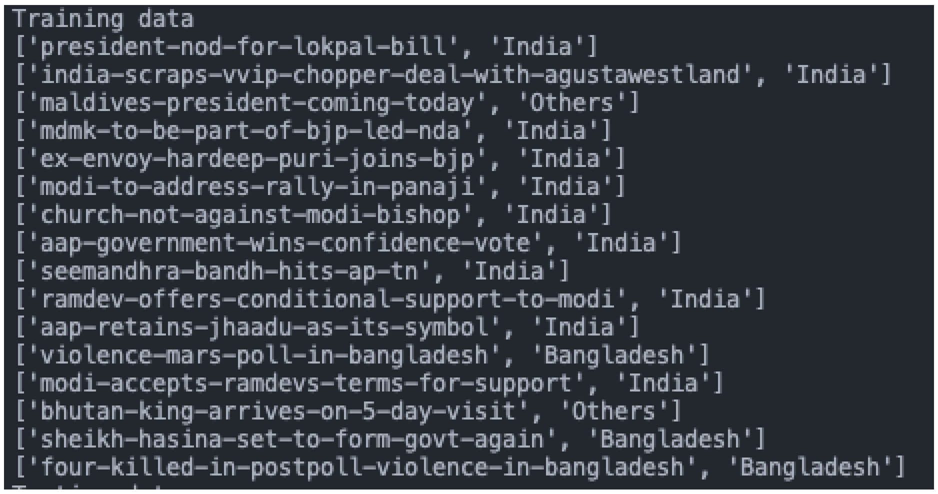
Code:

Reading and separating datasets:



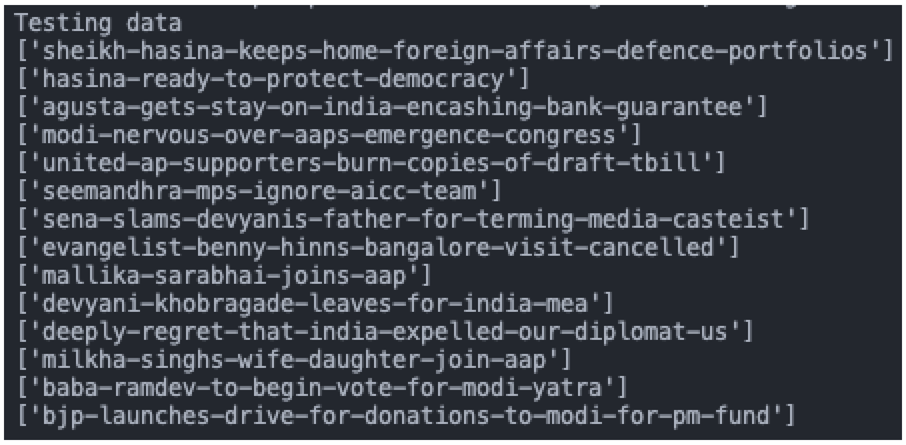
```
def seperate_datasets(self):
    training_flag=-1
    print('\nRecords of the dataset')
    for row in self.dataset_:
        if row[0]=="training data":
            training_flag=1
        if row[0]=="test data":
            training_flag=0

        if training_flag==-1:
            continue

        if training_flag==1:
            self.training_dataset.append([row[1],row[2]])
        else:
            self.testing_dataset.append([row[1]])

        print(row)
    print('')
```

Training data:



```
Training data
['president-nod-for-lokpal-bill', 'India']
['india-scraps-vvip-chopper-deal-with-agustawestland', 'India']
['maldives-president-coming-today', 'Others']
['mdmk-to-be-part-of-bjp-led-nda', 'India']
['ex-envoy-hardeep-puri-joins-bjp', 'India']
['modi-to-address-rally-in-panaji', 'India']
['church-not-against-modi-bishop', 'India']
['aap-government-wins-confidence-vote', 'India']
['seemandhra-bandh-hits-ap-tn', 'India']
['ramdev-offers-conditional-support-to-modi', 'India']
['aap-retains-jhaadu-as-its-symbol', 'India']
['violence-mars-poll-in-bangladesh', 'Bangladesh']
['modi-accepts-ramdevs-terms-for-support', 'India']
['bhutan-king-arrives-on-5-day-visit', 'Others']
['sheikh-hasina-set-to-form-govt-again', 'Bangladesh']
['four-killed-in-postpoll-violence-in-bangladesh', 'Bangladesh']
```

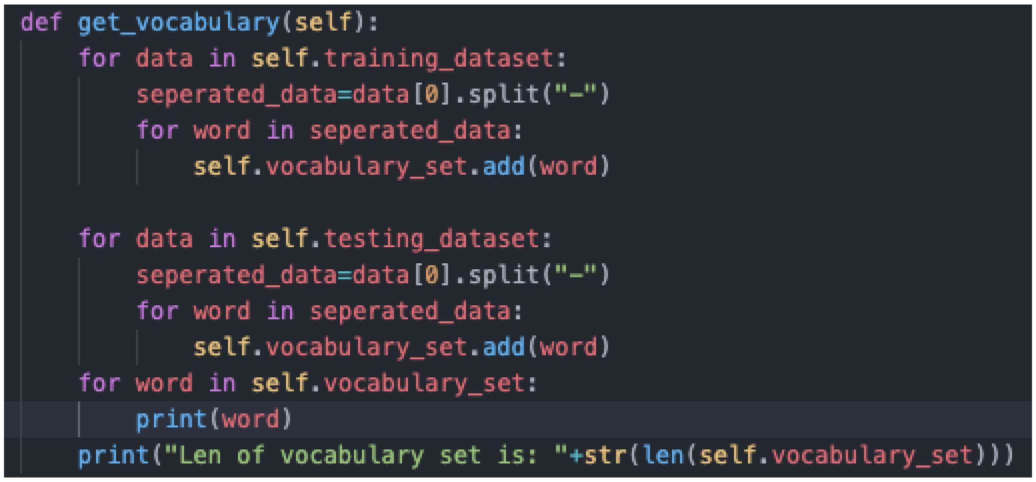
Testing data:



```
Testing data
['sheikh-hasina-keeps-home-foreign-affairs-defence-portfolios']
['hasina-ready-to-protect-democracy']
['agusta-gets-stay-on-india-encashing-bank-guarantee']
['modi-nervous-over-aaps-emergence-congress']
['united-ap-supporters-burn-copies-of-draft-tbill']
['seemandhra-mps-ignore-aicc-team']
['sena-slams-devyanis-father-for-terming-media-casteist']
['evangelist-benny-hinns-bangalore-visit-cancelled']
['mallika-sarabhai-joins-aap']
['devyani-khobragade-leaves-for-india-mea']
['deeply-regret-that-india-expelled-our-diplomat-us']
['milkha-singhs-wife-daughter-join-aap']
['baba-ramdev-to-begin-vote-for-modi-yatra']
['bjp-launches-drive-for-donations-to-modi-for-pm-fund']
```

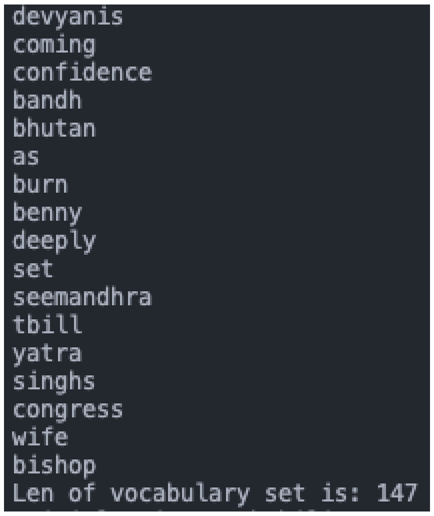
2. Find out all the unique words in **training and test data** to find all the vocabulary contents. Find the size of vocabulary |V| first.

Code:



```python
def get_vocabulary(self):
    for data in self.training_dataset:
        seperated_data=data[0].split("-")
        for word in seperated_data:
            self.vocabulary_set.add(word)

    for data in self.testing_dataset:
        seperated_data=data[0].split("-")
        for word in seperated_data:
            self.vocabulary_set.add(word)
    for word in self.vocabulary_set:
        print(word)
    print("Len of vocabulary set is: "+str(len(self.vocabulary_set)))
```

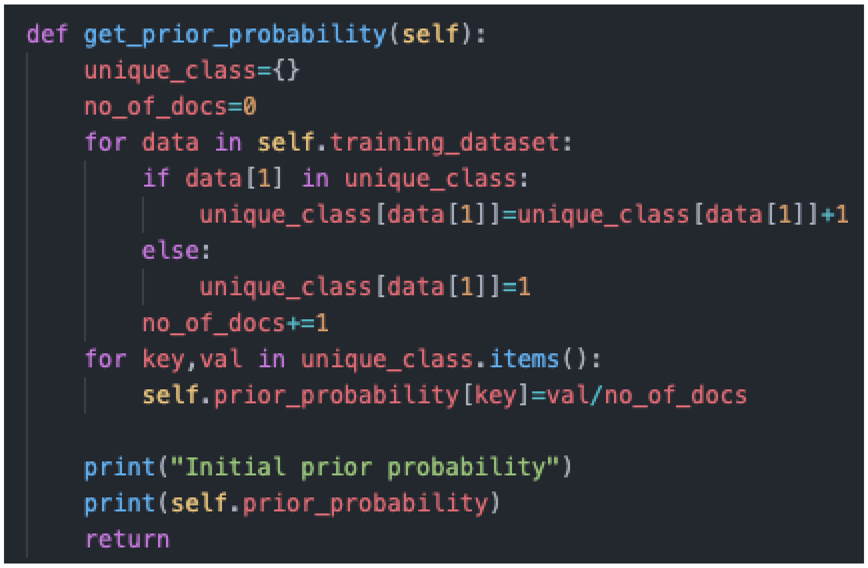
output:



```
devyanis
coming
confidence
bandh
bhutan
as
burn
benny
deeply
set
seemandhra
tbill
yatra
singhs
congress
wife
bishop
Len of vocabulary set is: 147
```

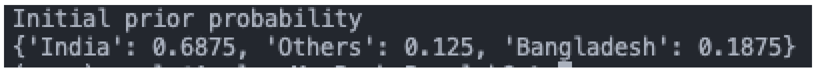
3 Find the number of documents belonging to an individual class in training data alone. Divide by the total number of documents to get the prior probability of that class.

Code:



```python
def get_prior_probability(self):
    unique_class={}
    no_of_docs=0
    for data in self.training_dataset:
        if data[1] in unique_class:
            unique_class[data[1]]=unique_class[data[1]]+1
        else:
            unique_class[data[1]]=1
        no_of_docs+=1
    for key,val in unique_class.items():
        self.prior_probability[key]=val/no_of_docs

    print("Initial prior probability")
    print(self.prior_probability)
    return
```

OutPut:



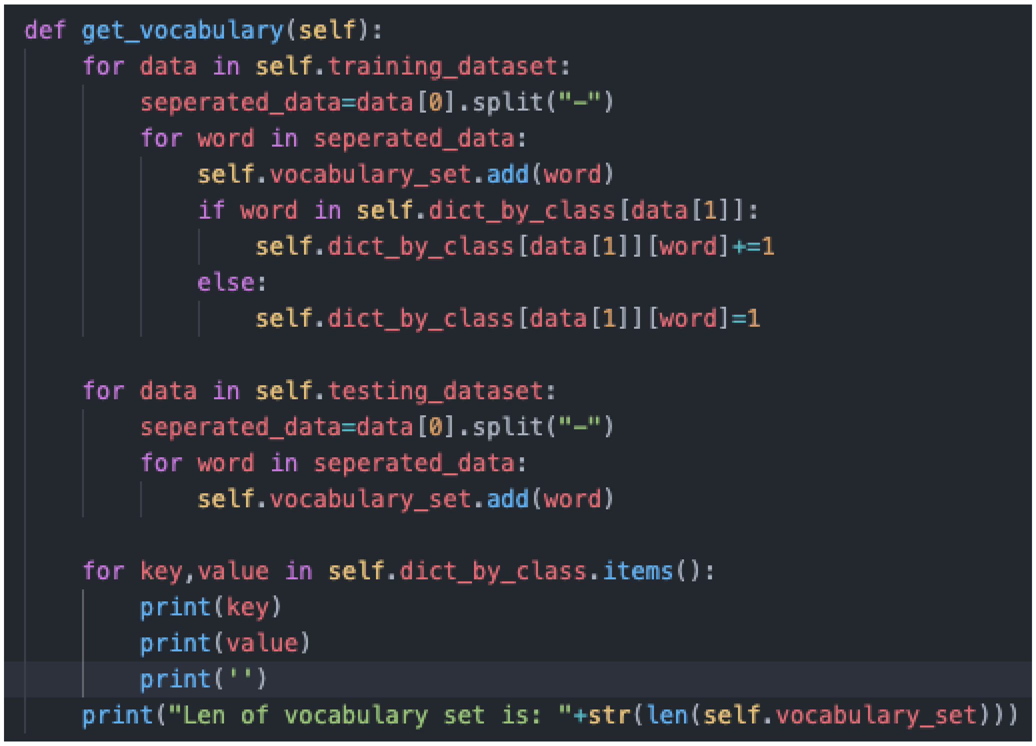
```
Initial prior probability
{'India': 0.6875, 'Others': 0.125, 'Bangladesh': 0.1875}
```

4. If you add all prior class probabilities it will come to 1.

P(India)+P(Others)+P(Bangladesh) = 0.6875+0.125+0.1875 =1

5. Now form a dictionary for every class containing all the words and their frequency.

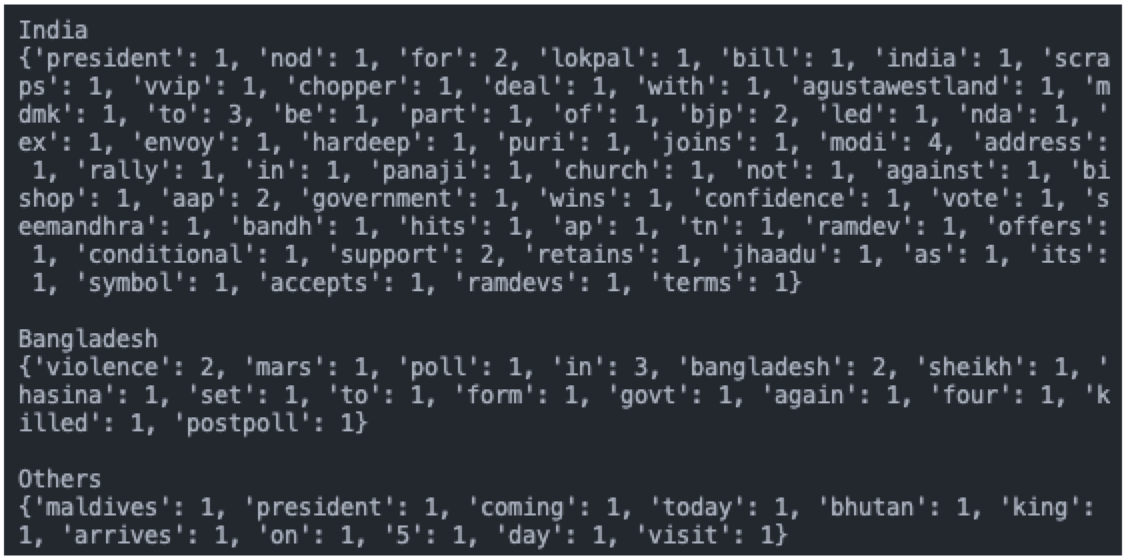
Code:



```python
def get_vocabulary(self):
    for data in self.training_dataset:
        seperated_data=data[0].split("-")
        for word in seperated_data:
            self.vocabulary_set.add(word)
            if word in self.dict_by_class[data[1]]:
                self.dict_by_class[data[1]][word]+=1
            else:
                self.dict_by_class[data[1]][word]=1

    for data in self.testing_dataset:
        seperated_data=data[0].split("-")
        for word in seperated_data:
            self.vocabulary_set.add(word)

    for key,value in self.dict_by_class.items():
        print(key)
        print(value)
        print('')
    print("Len of vocabulary set is: "+str(len(self.vocabulary_set)))
```

Output:

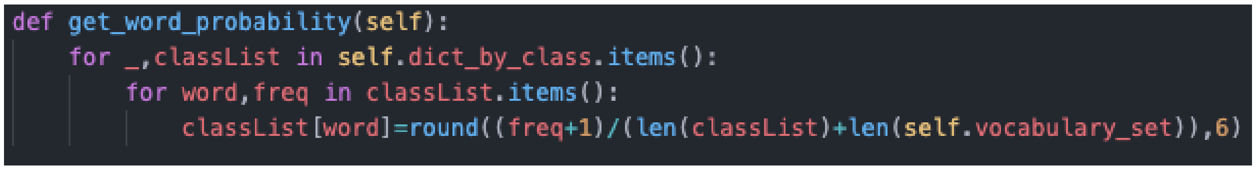
Dict by class with words and frequency



```
India
{'president': 1, 'nod': 1, 'for': 2, 'lokpal': 1, 'bill': 1, 'india': 1, 'scra
ps': 1, 'vvip': 1, 'chopper': 1, 'deal': 1, 'with': 1, 'agustawestland': 1, 'm
dmk': 1, 'to': 3, 'be': 1, 'part': 1, 'of': 1, 'bjp': 2, 'led': 1, 'nda': 1, '
ex': 1, 'envoy': 1, 'hardeep': 1, 'puri': 1, 'joins': 1, 'modi': 4, 'address':
 1, 'rally': 1, 'in': 1, 'panaji': 1, 'church': 1, 'not': 1, 'against': 1, 'bi
shop': 1, 'aap': 2, 'government': 1, 'wins': 1, 'confidence': 1, 'vote': 1, 's
eemandhra': 1, 'bandh': 1, 'hits': 1, 'ap': 1, 'tn': 1, 'ramdev': 1, 'offers':
 1, 'conditional': 1, 'support': 2, 'retains': 1, 'jhaadu': 1, 'as': 1, 'its':
 1, 'symbol': 1, 'accepts': 1, 'ramdevs': 1, 'terms': 1}

Bangladesh
{'violence': 2, 'mars': 1, 'poll': 1, 'in': 3, 'bangladesh': 2, 'sheikh': 1, '
hasina': 1, 'set': 1, 'to': 1, 'form': 1, 'govt': 1, 'again': 1, 'four': 1, 'k
illed': 1, 'postpoll': 1}

Others
{'maldives': 1, 'president': 1, 'coming': 1, 'today': 1, 'bhutan': 1, 'king':
1, 'arrives': 1, 'on': 1, '5': 1, 'day': 1, 'visit': 1}
```

6. Count the total number of words for every class.

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

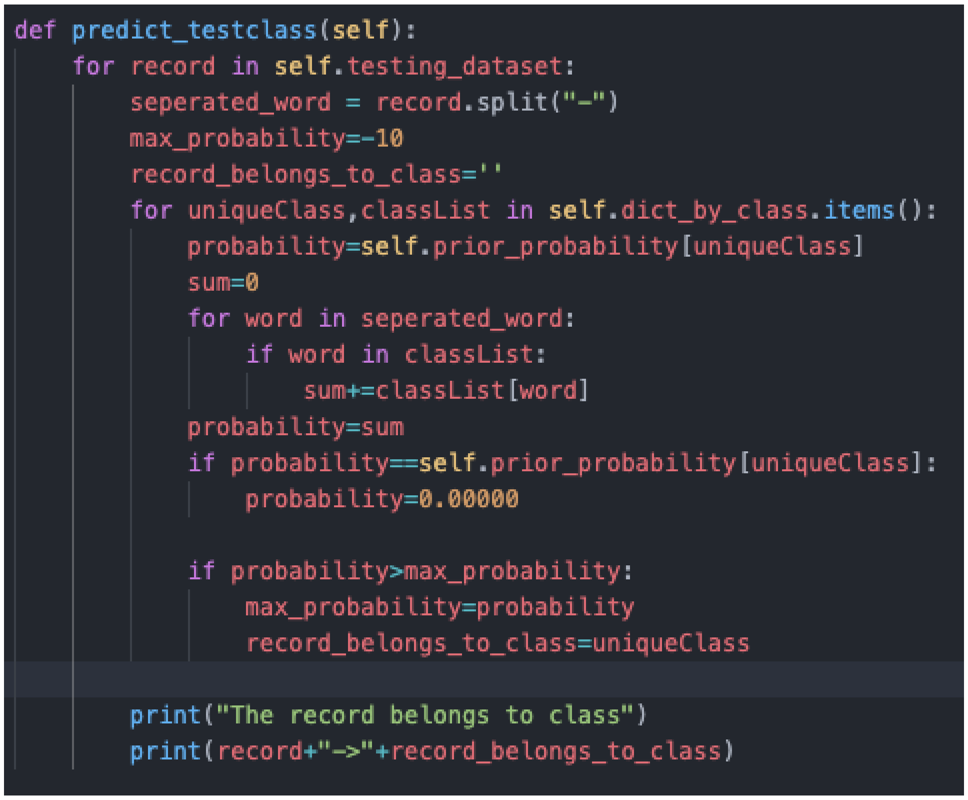


```python
def get_word_probability(self):
    for _,classList in self.dict_by_class.items():
        for word,freq in classList.items():
            classList[word]=round((freq+1)/(len(classList)+len(self.vocabulary_set)),6)
```

8. Now determine the probability of the chosen test document belonging to a class

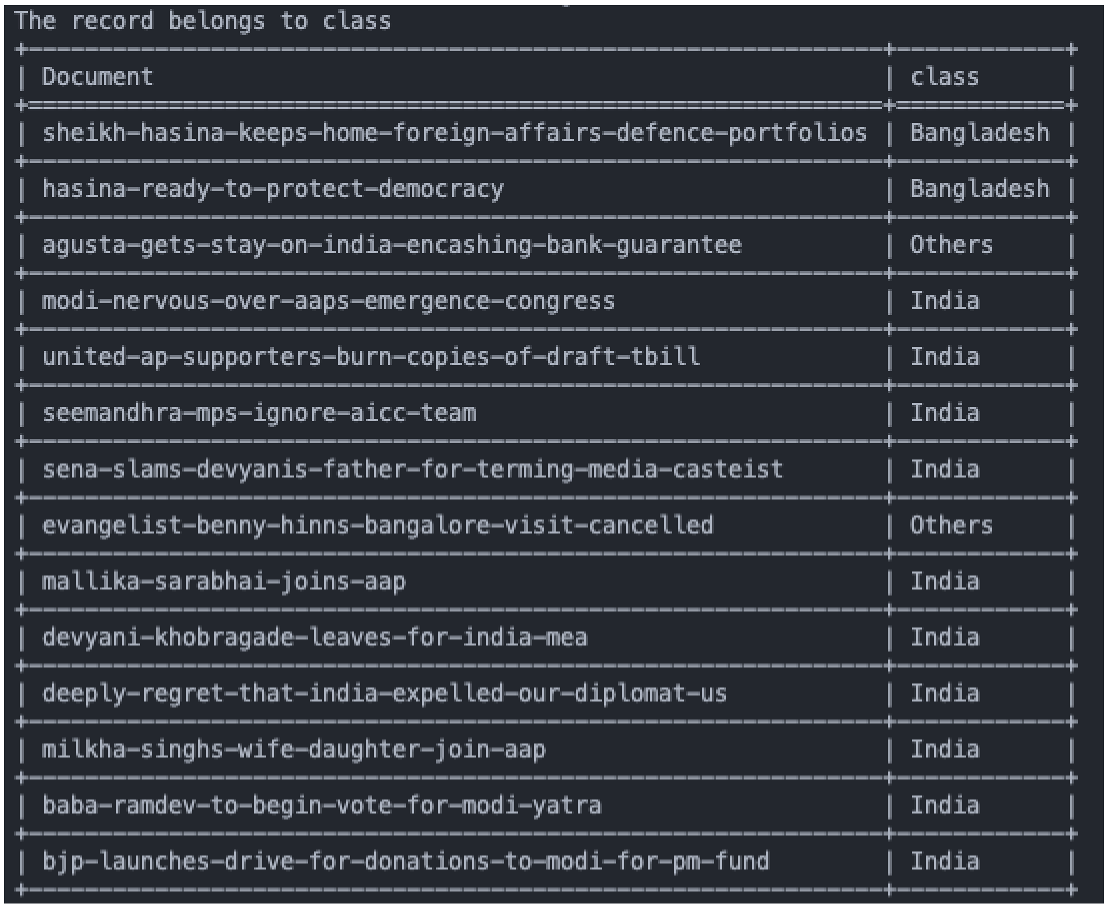
$$\hat{P}(c)\prod_{i}\hat{P}(x_i \mid c)$$

Code:



```python
def predict_testclass(self):
    for record in self.testing_dataset:
        seperated_word = record.split("-")
        max_probability=-10
        record_belongs_to_class=''
        for uniqueClass,classList in self.dict_by_class.items():
            probability=self.prior_probability[uniqueClass]
            sum=0
            for word in seperated_word:
                if word in classList:
                    sum+=classList[word]
            probability=sum
            if probability==self.prior_probability[uniqueClass]:
                probability=0.00000

            if probability>max_probability:
                max_probability=probability
                record_belongs_to_class=uniqueClass

        print("The record belongs to class")
        print(record+"->"+record_belongs_to_class)
```

Output:



```
The record belongs to class
+-------------------------------------------------------------+------------+
| Document                                                    | class      |
+=============================================================+============+
| sheikh-hasina-keeps-home-foreign-affairs-defence-portfolios | Bangladesh |
+-------------------------------------------------------------+------------+
| hasina-ready-to-protect-democracy                           | Bangladesh |
+-------------------------------------------------------------+------------+
| agusta-gets-stay-on-india-encashing-bank-guarantee          | Others     |
+-------------------------------------------------------------+------------+
| modi-nervous-over-aaps-emergence-congress                   | India      |
+-------------------------------------------------------------+------------+
| united-ap-supporters-burn-copies-of-draft-tbill             | India      |
+-------------------------------------------------------------+------------+
| seemandhra-mps-ignore-aicc-team                             | India      |
+-------------------------------------------------------------+------------+
| sena-slams-devyanis-father-for-terming-media-casteist       | India      |
+-------------------------------------------------------------+------------+
| evangelist-benny-hinns-bangalore-visit-cancelled            | Others     |
+-------------------------------------------------------------+------------+
| mallika-sarabhai-joins-aap                                  | India      |
+-------------------------------------------------------------+------------+
| devyani-khobragade-leaves-for-india-mea                     | India      |
+-------------------------------------------------------------+------------+
| deeply-regret-that-india-expelled-our-diplomat-us           | India      |
+-------------------------------------------------------------+------------+
| milkha-singhs-wife-daughter-join-aap                        | India      |
+-------------------------------------------------------------+------------+
| baba-ramdev-to-begin-vote-for-modi-yatra                    | India      |
+-------------------------------------------------------------+------------+
| bjp-launches-drive-for-donations-to-modi-for-pm-fund        | India      |
+-------------------------------------------------------------+------------+
```

Though 2 times the model has predicted wrong results. The rest of the data was predicted accurately.

Accuracy:12/14*100 = 86%