```
# importing necessary libraries

import pandas as pd  #we can do data manipulatin and analysis
import numpy as np   #to do numerical computing


# Loading the dataset
df = pd.read_csv('/content/SMSSpamCollection.csv',sep='\t',names = ['label','message'])
df.head() # This function will show top 5 rows of dataset
```

	label	message
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

```
# This function will show the no.of rows and columns of dataset
df.shape
```

```
(5572, 2)
```

```
# This function will show column names of dataset
df.columns
```

```
Index(['label', 'message'], dtype='object')
```

```
# NLTK is a basic NLP library, used to deal human language data and to perform text processing
# re is used to deal regular expressions and text manipulation, pattern matching, data extraction
import nltk
import re

# This code will download the stopwords(stopwords=words which does not have any particular meaning)
from nltk.corpus import stopwords
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
True
```

```
# importing porterstemmer to get root word, but stemming will not give meaningful words
from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()


# Creating a null corpus
# removing spaces, symbols and lowering the words and splitting the each sentence
# and performing the stemming process on 'message' column of datset
# finally adding the cleaned data to the null corpus
corpus = []
for i in range(0,len(df)):
  dataclean = re.sub('[^a-zA-Z]'," ",df['message'][i])
  dataclean = dataclean.lower()
  dataclean = dataclean.split()
  dataclean = [ps.stem(word) for word in dataclean if not word in stopwords.words('english')]
  dataclean =" ".join(dataclean)
  corpus.append(dataclean)


#it will show the data stored in corpus
corpus
```

```
 'gudnit tc practic go',
 'di yiju ju saw ur mail case huim havent sent u num di num',
 'one small prestig problem',
 'fanci shag interest sextextuk com txt xxuk suzi txt cost per msg tnc websit x',
 'check realli miss see jeremiah great month',
 'nah help never iphon',
 'car hour half go apeshit',
 'today sorri day ever angri ever misbehav hurt plz plz slap urself bcoz ur fault basic good',
 'yo guy ever figur much need alcohol jay tri figur much safe spend weed',
 'lt gt ish minut minut ago wtf',
 'thank call forgot say happi onam sirji fine rememb met insur person meet qatar insha allah rakhesh ex tata aig join tissco
tayseer',
 'congratul ur award cd voucher gift guarante free entri wkli draw txt music tnc www ldew com win ppmx age',
 'ur cash balanc current pound maxim ur cash send cash p msg cc hg suit land row w j hl',
 'actor work work even sleep late sinc unemploy moment alway sleep late unemploy everi day saturday',
 'hello got st andrew boy long way cold keep post',
 'ha ha cool cool chikku chikku db',
 'oh ok prob',
 'check audrey statu right',
 'busi tri finish new year look forward final meet',
 'good afternoon sunshin dawn day refresh happi aliv breath air smile think love alway',
 'well know z take care worri',
 'updat xma offer latest motorola sonyericsson nokia free bluetooth doubl min txt orang call mobileupd call optout f q',
 'discount code rp stop messag repli stop www regalportfolio co uk custom servic',
 'wat uniform get',
 'cool text readi',
 'hello boytoy geeee miss alreadi woke wish bed cuddl love',
 'spoil bed well',
 'go bath msg next lt gt min',
 'cant keep talk peopl sure pay agre price pl tell want realli buy much will pay',
 'thank rington order refer charg gbp per week unsubscrib anytim call custom servic',
 'say happen',
 'could seen recognis face',
 'well lot thing happen lindsay new year sigh bar ptbo blue heron someth go',
 'keep payasam rinu bring',
 'taught ranjith sir call sm like becau he verifi project prabu told today pa dont mistak',
 'guess worri must know way bodi repair quit sure worri take slow first test guid ovul relax noth said reason worri keep followin',
 'yeah sure give coupl minut track wallet',
 'hey leav big deal take care',
 'hey late ah meet',
 'doubl min txt month free bluetooth orang avail soni nokia motorola phone call mobileupd call optout n dx',
 'took mr owl lick',
 'custom place call',
 'mm time dont like fun',
 'mth half price orang line rental latest camera phone free phone mth call mobilesdirect free updat stoptxt',
 'yup lunch buffet u eat alreadi',
 'huh late fr dinner',
 'hey sat go intro pilat kickbox',
 'morn ok'
```

```
# tfidf vectorizer represent text data as numerical features
from sklearn .feature_extraction.text import TfidfVectorizer
tf = TfidfVectorizer(max_features = 2500)
X = tf.fit_transform(corpus).toarray()
```

```
X.shape
```

```
(5572, 2500)
```

```
#This fuction will replace zero and one as dummy values in the 'label' column
y = pd.get_dummies(df['label'])
```

```
#splitting the dataset
y = df.iloc[:,1].values
```

```
#Splitting the dataset to train and test
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split( X,y,test_size = 0.3,random_state = 42)
```

```
#importing multinomial naive bayes Algorithm to understand and predict
from sklearn.naive_bayes import MultinomialNB
nb = MultinomialNB()
model = nb.fit(X_train, y_train)  #this function is used to train the dataset
y_pred = model.predict(X_test)    #predict model
```

```
#importing metrics to know the performance of model
from sklearn.metrics import accuracy_score, confusion_matrix
```

```
accuracy_score(y_test,y_pred)
```

```
0.01076555023923445
```

```
confusion_matrix(y_test, y_pred)
```

```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]])
```

```
# This code will download the stopwords(stopwords=words which does not have any particular meaning)
## importing lemmatizerr to get root word,lemmatization will give meaningful words
from nltk.stem import WordNetLemmatizer
nltk.download('wordnet')
ln =WordNetLemmatizer()
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
```

```
# Creating a null corpus
# removing spaces, symbols and lowering the words and splitting the each sentence
# and performing the lemmatization process on 'message' column of datset
# finally adding the cleaned data to the null corpus
corpus = []
for i in range(0,len(df)):
  dataclean = re.sub('[^a-zA-Z]'," ",df['message'][i])
  dataclean = dataclean.lower()
  dataclean = dataclean.split()
  dataclean = [ln.lemmatize(word) for word in dataclean if not word in stopwords.words('english')]
  dataclean =" ".join(dataclean)
  corpus.append(dataclean)
```

```
#it will show the data stored in dataset
corpus
```

```
        'watching telugu movie wat abt u',
        'see finish load loan pay',
        'hi wk ok hols yes bit run forgot hairdresser appointment four need get home n shower beforehand cause prob u',
        'see cup coffee animation',
        'please text anymore nothing else say',
        'okay name ur price long legal wen pick u ave x am xx',
        'still looking car buy gone driving test yet',
        'per request melle melle oru minnaminunginte nurungu vettam set callertune caller press copy friend callertune',
        'wow right mean guess gave boston men changed search location nyc something changed cuz signin page still say boston',
        'umma life vava umma love lot dear',
        'thanks lot wish birthday thanks making birthday truly memorable',
        'aight hit get cash',
        'would ip address test considering computer minecraft server',
        'know grumpy old people mom like better lying always one play joke',
        'dont worry guess busy',
        'plural noun research',
        'going dinner msg',
        'ok wif co like try new thing scared u dun like mah co u said loud',
```

```
# creating Tfidf vectorizer
# tfidf vectorizer represent text data as numerical features
from sklearn.feature_extraction.text import TfidfVectorizer
tf = TfidfVectorizer(max_features=2500)
X=tf.fit_transform(corpus).toarray()
```

```
y = pd.get_dummies(df['label'])
```

```
y = y.iloc[:,1].values
```

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split( X, y,test_size = 0.3,random_state = 42)
```

```
from sklearn.naive_bayes import MultinomialNB
nb = MultinomialNB()
model = nb.fit(X_train, y_train)
```

```
y_pred = model.predict(X_test)
```

```
from sklearn.metrics import accuracy_score, confusion_matrix
accuracy_score(y_test,y_pred)
```

```
0.9808612440191388
```

```
confusion_matrix(y_test,y_pred)
```

```
array([[1446,    2],
       [  30,  194]])
```