Oasis Infobyte Internship

Intern Name -Akshay Anandkar

Task 4-EMAIL SPAM DETECTION WITH MACHINE LEARNING

Problem Statement- We've all been the recipient of spam emails before. Spam mail, or junk mail, is a type of email that is sent to a massive number of users at one time, frequently containing cryptic messages, scams, or most dangerously, phishing content.In this Project, use Python to build an email spam detector. Then, use machine learning to train the spam detector to recognize and classify emails into spam and non-spam. Let's getstarted!

In [1]:
```
#import all require liabrabries
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

In [2]:
```
#importing dataset
email=pd.read_csv(r"D:\Data-Science-Internship\spam.csv",encoding='ISO-8859-1')
email
```

Out[2]:

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN
...	...	...	...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...	NaN	NaN	NaN
5568	ham	Will Ì_ b going to esplanade fr home?	NaN	NaN	NaN
5569	ham	Pity, * was in mood for that. So...any other s...	NaN	NaN	NaN
5570	ham	The guy did some bitching but I acted like i'd...	NaN	NaN	NaN
5571	ham	Rofl. Its true to its name	NaN	NaN	NaN

5572 rows × 5 columns

In [3]:
```
email.shape
```

Out[3]:
```
(5572, 5)
```

In [4]:
```
email.isnull().sum()
```

Out[4]:
```
v1              0
v2              0
Unnamed: 2   5522
Unnamed: 3   5560
Unnamed: 4   5566
dtype: int64
```

In [5]:
```
#replacing a null values with null string
email_data=email.where(pd.notnull(email),'')
```

In [6]:
```
email_data.isnull().sum()
```

Out[6]:
```
v1            0
v2            0
Unnamed: 2    0
Unnamed: 3    0
Unnamed: 4    0
dtype: int64
```

In [7]:
```
#removing all unecessary columns
email.drop(columns=['Unnamed: 2','Unnamed: 3','Unnamed: 4'],inplace=True)
```

In [8]:
```
email.head()
```

Out[8]:

	v1	v2
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...

In [9]:
```
#remane v1 and v2 columns as category and message
email=email.rename(columns={'v1': 'Category','v2': 'Message'})
email.head(10)
```

Out[9]:

	Category	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...
5	spam	FreeMsg Hey there darling it's been 3 week's n...
6	ham	Even my brother is not like to speak with me. ...
7	ham	As per your request 'Melle Melle (Oru Minnamin...
8	spam	WINNER!! As a valued network customer you have...
9	spam	Had your mobile 11 months or more? U R entitle...

In [10]:
```
#Applyin label encoding further spam as 0 and ham as 1
email.loc[email['Category']=='spam','Category']=0
email.loc[email['Category']=='ham','Category']=1
email.head()
```

Out[10]:

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

In [11]:
```
#Getting Independent and dependent varaibles
X=email['Message']
y=email['Category']
```

In [12]:
```
print(X)
```

```
0       Go until jurong point, crazy.. Available only ...
1                           Ok lar... Joking wif u oni...
2       Free entry in 2 a wkly comp to win FA Cup fina...
3       U dun say so early hor... U c already then say...
4       Nah I don't think he goes to usf, he lives aro...
                              ...
5567    This is the 2nd time we have tried 2 contact u...
5568                Will Ì_ b going to esplanade fr home?
5569    Pity, * was in mood for that. So...any other s...
5570    The guy did some bitching but I acted like i'd...
5571                           Rofl. Its true to its name
Name: Message, Length: 5572, dtype: object
```

In [13]:
```
print(y)
```

```
0       1
1       1
2       0
3       1
4       1
       ..
5567    0
5568    1
5569    1
5570    1
5571    1
Name: Category, Length: 5572, dtype: object
```

In [14]:
```
#Splitting data into train test split for model building
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=3)
```

In [15]:
```
print(X.shape)
print(X_train.shape)
print(X_test.shape)
```

```
(5572,)
(4457,)
(1115,)
```

In [16]:
```
print(y.shape)
print(y_train.shape)
print(y_test.shape)
```

```
(5572,)
(4457,)
(1115,)
```

In [17]:
```
#Using feature extraction for further analysis
feature_extraction=TfidfVectorizer(min_df=1,stop_words='english',lowercase=True)
```

In [18]:
```
X_train_features=feature_extraction.fit_transform(X_train)
X_test_features=feature_extraction.transform(X_test)
```

In [19]:
```
#now convert y_train and y_test into integer data
y_train=y_train.astype('int')
y_test=y_test.astype('int')
```

In [20]:
```
print(X_train)
```

```
3075    Mum, hope you are having a great day. Hoping t...
1787                           Yes:)sura in sun tv.:)lol.
1614    Me sef dey laugh you. Meanwhile how's my darli...
4304                Yo come over carlos will be here soon
3266                    Ok then i come n pick u at engin?
                              ...
789                          Gud mrng dear hav a nice day
968              Are you willing to go for aptitude class.
1667    So now my dad is gonna call after he gets out ...
3321    Ok darlin i supose it was ok i just worry too ...
1688                     Nan sonathaya soladha. Why boss?
Name: Message, Length: 4457, dtype: object
```

In [21]:
```
print(X_train_features)
```

```
  (0, 741)	0.3219352588930141
  (0, 3979)	0.2410582143632299
  (0, 4296)	0.3891385935794867
  (0, 6599)	0.20296878731699391
  (0, 3386)	0.3219352588930141
  (0, 2122)	0.38613577623520473
  (0, 3136)	0.440116181574609
  (0, 3262)	0.25877035357606315
  (0, 3380)	0.21807195185332803
  (0, 4513)	0.2909649098524696
  (1, 4061)	0.380431198316959
  (1, 6872)	0.4306015894277422
  (1, 6417)	0.4769136859540388
  (1, 6442)	0.5652509076654626
  (1, 7443)	0.35056971070320353
  (2, 933)	0.4917598465723273
  (2, 2109)	0.42972812260098503
  (2, 3917)	0.40088501350982736
  (2, 2226)	0.413484525934624
  (2, 5825)	0.4917598465723273
  (3, 6140)	0.4903863168693604
  (3, 1599)	0.5927091854194291
  (3, 1842)	0.3708680641487708
  (3, 7453)	0.5202633571003087
  (4, 2531)	0.7419319091456392
  :	:
  (4452, 2122)	0.31002103760284144
  (4453, 999)	0.6760129013031282
  (4453, 7273)	0.5787739591782677
  (4453, 1762)	0.45610005640082985
  (4454, 3029)	0.42618909997886
  (4454, 2086)	0.3809693742808703
  (4454, 3088)	0.34475593009514444
  (4454, 2001)	0.4166919007849217
  (4454, 1049)	0.31932060116006045
  (4454, 7346)	0.31166263834107377
  (4454, 5370)	0.42618909997886
  (4455, 1148)	0.38998123077430413
  (4455, 6433)	0.38998123077430413
  (4455, 6361)	0.25697343671652706
  (4455, 2764)	0.3226323745940581
  (4455, 7358)	0.2915949626395065
  (4455, 7407)	0.3028481995557642
  (4455, 2108)	0.3136468384526087
  (4455, 4251)	0.30616657078392584
  (4455, 3763)	0.16807158405536876
  (4455, 4773)	0.35860460546223444
  (4456, 6117)	0.5304350313291551
  (4456, 6133)	0.5304350313291551
  (4456, 1386)	0.4460036316446079
  (4456, 4557)	0.48821933148688146
```

Training the model

In [22]:
```
#here we are using logistic regression
model= LogisticRegression()
model.fit(X_train_features,y_train)
```

Out[22]:
```
▾ LogisticRegression
LogisticRegression()
```

In [30]:
```
#Evaluation of the training model
y_train_pred=model.predict(X_train_features)
accuray_on_train_data=accuracy_score(y_train,y_train_pred)
```

In [31]:
```
print("Accuray for train data is:",accuray_on_train_data)
```

```
Accuray for train data is: 0.9661207089970832
```

In [32]:
```
y_pred=model.predict(X_test_features)
accuray_on_test_data=accuracy_score(y_test,y_pred)
```

In [33]:
```
print("Accuracy for test data",accuray_on_test_data)
```

```
Accuracy for test data 0.9623318385650225
```

In [36]:
```
#prediction for sample input mail
input_user_email=["Java/Asp.net/Design Engineer/Software Testing/PHP/web/Networking/Software Developer/Python,Angular, Data Scientist,Salesforce,Hac
input_data_features=feature_extraction.transform(input_user_email)
```

In [40]:
```
prediction=model.predict(input_data_features)
print(prediction)
```

```
[1]
```

In [41]:
```
if prediction==1:
    print("it is ham mail")
else:
    print("It is spam mail")
```

```
it is ham mail
```

In [ ]: