Email Spam Classifier

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import  accuracy_score
```

Tfidfvectorizer :-

```
    1. It Is a feature of extraction technique commonly used in
natural processing NLP and text mining tasks.

    2. In easy Words It converts the text documents into a numerical
representation that a machine learning algorithm can understand and
work with.
```

Logistic Regression Model :-

```
    1. It is a Very Popular Classification Algorithm.It is a Part of
scikitlearn Lib.in python.
    2. It is sutaible for Binary Classification Problems where the
target variable has 2 classes.
```

Accuracy Score :-

```
    The Accuracy score function is a performance metric provided by
the scikit learn model. It is used to calcilate the Accuracy of ML
models
```

```
df=pd.read_csv(r"C:\Users\hp\Desktop\Atharva DA\Data-Science-Projects\
Email Spam Classifier\mail_data.csv")
```

```
df
```

```
     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...
...       ...                                                ...
5567     spam  This is the 2nd time we have tried 2 contact u...
5568      ham               Will ü b going to esplanade fr home?
5569      ham  Pity, * was in mood for that. So...any other s...
5570      ham  The guy did some bitching but I acted like i'd...
5571      ham                         Rofl. Its true to its name
```

```
[5572 rows x 2 columns]

data=df.where((pd.notnull(df)), '')

data.head()

  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...

data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 2 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Category  5572 non-null   object
 1   Message   5572 non-null   object
dtypes: object(2)
memory usage: 87.2+ KB

data.shape

(5572, 2)

data.loc[data['Category']=='spam','Category',] = 0
data.loc[data["Category"] == "ham","Category",] = 1
```

0---- Spam

1---- Ham

```
data

     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...
...       ...                                                ...
5567        0  This is the 2nd time we have tried 2 contact u...
5568        1               Will ü b going to esplanade fr home?
5569        1  Pity, * was in mood for that. So...any other s...
5570        1  The guy did some bitching but I acted like i'd...
5571        1                         Rofl. Its true to its name
```

```
[5572 rows x 2 columns]

X = data['Message']
Y = data['Category']

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 ü 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

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

x_train,X_test,Y_train,Y_test =
train_test_split(X,Y,test_size=0.2,random_state=3)
```

Randome State :-

```
Randome State is a hyper parameter that is used to control any such
randomeness involved in machine learning model to get consistent
result
it is used to help the processes of centroid clustering

print(X.shape)
print(X_test.shape)
print(x_train.shape)
```

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

```
print(Y.shape)
print(Y_test.shape)
print(Y_train.shape)
```

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

```
feature_extraction = TfidfVectorizer(min_df=1, stop_words='english',
lowercase=True)
X_train_features = feature_extraction.fit_transform(x_train)
X_test_features = feature_extraction.transform(X_test)


Y_train = Y_train.astype('int')
Y_test = Y_test.astype('int')
```

```
print(x_train
      )
```

```
3075                      Don know. I did't msg him recently.
1787    Do you know why god created gap between your f...
1614                             Thnx dude. u guys out 2nite?
4304                                          Yup i'm free...
3266    44 7732584351, Do you want a New Nokia 3510i c...
                              ...                        
789     5 Free Top Polyphonic Tones call 087018728737,...
968     What do u want when i come back?.a beautiful n...
1667    Guess who spent all last night phasing in and ...
3321    Eh sorry leh... I din c ur msg. Not sad alread...
1688    Free Top ringtone -sub to weekly ringtone-get ...
Name: Message, Length: 4457, dtype: object
```

```
print(X_train_features)
```

```
<Compressed Sparse Row sparse matrix of dtype 'float64'
     with 34775 stored elements and shape (4457, 7431)>
  Coords   Values
  (0, 2329)     0.38783870336935383
  (0, 3811)     0.34780165336891333
  (0, 2224)     0.413103377943378
  (0, 4456)     0.4168658090846482
  (0, 5413)     0.6198254967574347
  (1, 3811)     0.17419952275504033
  (1, 3046)     0.2503712792613518
  (1, 1991)     0.33036995955537024
  (1, 2956)     0.33036995955537024
  (1, 2758)	0.3226407885943799
  (1, 1839)	0.2784903590561455
  (1, 918)	0.22871581159877646
  (1, 2746)	0.3398297002864083
  (1, 2957)	0.3398297002864083
  (1, 3325)	0.31610586766078863
  (1, 3185)	0.29694482957694585
  (1, 4080)	0.18880584110891163
  (2, 6601)	0.6056811524587518
  (2, 2404)	0.45287711070606745
  (2, 3156)	0.4107239318312698
  (2, 407)	0.509272536051008
  (3, 7414)	0.8100020912469564
  (3, 2870)	0.5864269879324768
  (4, 2870)	0.41872147309323743
  (4, 487)	0.2899118421746198
  :	:
  (4454, 2855)	0.47210665083641806
  (4454, 2246)	0.47210665083641806
  (4455, 4456)	0.24920025316220423
  (4455, 3922)	0.31287563163368587
  (4455, 6916)	0.19636985317119715
  (4455, 4715)	0.30714144758811196
  (4455, 3872)	0.3108911491788658
  (4455, 7113)	0.30536590342067704
  (4455, 6091)	0.23103841516927642
  (4455, 6810)	0.29731757715898277
  (4455, 5646)	0.33545678464631296
  (4455, 2469)	0.35441545511837946
  (4455, 2247)	0.37052851863170466
  (4456, 2870)	0.31523196273113385
  (4456, 5778)	0.16243064490100795
  (4456, 334)	0.2220077711654938
  (4456, 6307)	0.2752760476857975
  (4456, 6249)	0.17573831794959716
  (4456, 7150)	0.3677554681447669
  (4456, 7154)	0.24083218452280053
  (4456, 6028)	0.21034888000987115
  (4456, 5569)	0.4619395404299172
  (4456, 6311)	0.30133182431707617
  (4456, 647)	0.30133182431707617
  (4456, 141)	0.292943737785358
```

```
Model = LogisticRegression()
```

```
Model.fit(X_train_features, Y_train)
```

```
LogisticRegression()
```

```
prediction_on_training_data = Model.predict(X_train_features)
```

```
accuracy_on_training_data = accuracy_score(Y_train,
prediction_on_training_data)

print("Accuracy on training data: ", accuracy_on_training_data)

Accuracy on training data:  0.9676912721561588
```

This means the model has accuracy of 96.7 %

```
prediction_on_test_data = Model.predict(X_test_features)

accuracy_on_test_data = accuracy_score(Y_test,
prediction_on_test_data)

print("Accuracy on test data: ", accuracy_on_test_data)

Accuracy on test data:  0.9668161434977578
```

This is almost as same as tain data result

```
input_your_mail = [
    "Hey, I hope you are doing well. I wanted to let you know about an
amazing opportunity that could change your life. Click here to find
out more!",


]

input_data_features = feature_extraction.transform(input_your_mail)

prediction = Model.predict(input_data_features)

print(prediction)
for i in prediction:
    if prediction[0] == 1:
        print("Ham mail")
    else:
        print("Spam mail") ;

[1]
Ham mail

input_your_mail = [
    "Congratulations! You have won a free vacation to a tropical
paradise. Click here to claim your prize!",
]

input_data_features = feature_extraction.transform(input_your_mail)

prediction = Model.predict(input_data_features)
```

```python
print(prediction)
for i in prediction:
    if prediction[0] == 1:
        print("Ham mail")
    else:
        print("Spam mail")
```

```
[0]
Spam mail
```