Sakshi Kharat

Oasis Infobyte (Data Science) - Task-4

Email Spam Detection using Machine Learning

About the Task- 4: Spam mails or junk mails are sent to the massive number of users at the same time, frequently containing the spam messages, scams or most dangerously, phishing content. This project will use Machine Learning to recognize and Classify Emails into Spam and non-spam

```
import pandas as pd

import matplotlib as plt

import numpy as np
```

Data Collection

```
from google.colab import files


uploaded = files.upload()

<IPython.core.display.HTML object>

Saving spam.csv to spam.csv

df = pd.read_csv('spam.csv', encoding='latin-1')
```

Viewing the Dataset

```
df

     Category                                            Message
Unnamed: 2  \
0         ham  Go until jurong point, crazy.. Available only ...
NaN
1         ham                      Ok lar... Joking wif u oni...
NaN
2        spam  Free entry in 2 a wkly comp to win FA Cup fina...
NaN
3         ham  U dun say so early hor... U c already then say...
NaN
4         ham  Nah I don't think he goes to usf, he lives aro...
NaN
...       ...                                                ...
...
5567     spam  This is the 2nd time we have tried 2 contact u...
NaN
```

```
5568      ham              Will Ì_ b going to esplanade fr home?
NaN
5569      ham  Pity, * was in mood for that. So...any other s...
NaN
5570      ham  The guy did some bitching but I acted like i'd...
NaN
5571      ham                         Rofl. Its true to its name
NaN

     Unnamed: 3 Unnamed: 4
0           NaN        NaN
1           NaN        NaN
2           NaN        NaN
3           NaN        NaN
4           NaN        NaN
...         ...        ...
5567        NaN        NaN
5568        NaN        NaN
5569        NaN        NaN
5570        NaN        NaN
5571        NaN        NaN

[5572 rows x 5 columns]
```

Displaying the Dataset

```
df.info

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

```
NaN
5571      ham                         Rofl. Its true to its name
NaN

     Unnamed: 3 Unnamed: 4
0           NaN        NaN
1           NaN        NaN
2           NaN        NaN
3           NaN        NaN
4           NaN        NaN
...         ...        ...
5567        NaN        NaN
5568        NaN        NaN
5569        NaN        NaN
5570        NaN        NaN
5571        NaN        NaN

[5572 rows x 5 columns]>
```

Dropping the null values

```
df_cleaned = df.drop(["Unnamed: 2","Unnamed: 3","Unnamed: 4"], axis=1)

df_cleaned

     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 Ì_ 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]
```

There are 5572 rows and 2 columns

Data Preprocessing

```
df_cleaned.columns

Index(['Category', 'Message'], dtype='object')
```

There are total 2 columns in the Dataset

```
df_cleaned.shape
```

```
(5572, 2)
```

```
df_cleaned.head(2)
```

```
  Category                                            Message
0      ham  Go until jurong point, crazy.. Available only ...
1      ham                      Ok lar... Joking wif u oni...
```

Displaying the first 2 entries of the Dataset

```
df_cleaned.tail(2)
```

```
     Category                                            Message
5570      ham  The guy did some bitching but I acted like i'd...
5571      ham                         Rofl. Its true to its name
```

Displaying the last 2 entries of the Dataset

```
df_cleaned.iloc[1]
```

```
Category                              ham
Message     Ok lar... Joking wif u oni...
Name: 1, dtype: object
```

Preprocessing the Data

Checking the Null Values

```
df_cleaned.isnull()
```

```
      Category  Message
0        False    False
1        False    False
2        False    False
3        False    False
4        False    False
...        ...      ...
5567     False    False
5568     False    False
5569     False    False
5570     False    False
5571     False    False

[5572 rows x 2 columns]
```

```
df_cleaned.isnull().sum()
```

```
Category    0
Message     0
dtype: int64

df_cleaned.isna().sum()

Category    0
Message     0
dtype: int64
```

Label Encoding

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

df_cleaned.head(4)

  Category                                            Message  Spam
0        1  Go until jurong point, crazy.. Available only ...     0
1        1                      Ok lar... Joking wif u oni...     0
2        0  Free entry in 2 a wkly comp to win FA Cup fina...     1
3        1  U dun say so early hor... U c already then say...     0
```

Splitting the data into to training data and test data

```
from sklearn.model_selection import train_test_split

from sklearn.feature_extraction.text import CountVectorizer
```

Using CountVectorizer class from scikit-learn to convert the text data into a bag-of-words representation.

```
from sklearn.naive_bayes import MultinomialNB

from sklearn.metrics import accuracy_score

from sklearn import preprocessing

lab = preprocessing.LabelEncoder()

df_cleaned['Category'] = lab.fit_transform(df_cleaned['Category'])

<ipython-input-49-30359793f41d>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation:
https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#
returning-a-view-versus-a-copy
  df_cleaned['Category'] = lab.fit_transform(df_cleaned['Category'])
```

Droping the Duplicate Values

```
df_cleaned.duplicated().sum()
```

```
0
```

403 are the Duplicate Values

```
df_cleaned  = df_cleaned.drop_duplicates()
```

```
df_cleaned.duplicated().sum()
```

```
0
```

Splitting the data

```
x = df_cleaned['Message']
```

```
y = df_cleaned['Category']
```

```
x_train, x_test, y_train, y_test =train_test_split(x,y,
test_size=0.25, random_state=0)
```

Feature Extraxtion - converting text into numerical values

```
from sklearn.pipeline import Pipeline
```

```
clt=Pipeline([
    ('vectorizer',CountVectorizer()),
    ('nb',MultinomialNB())
])
```

Training the model

```
clt.fit(x_train,y_train)
```

```
Pipeline(steps=[('vectorizer', CountVectorizer()), ('nb',
MultinomialNB())])
```

```
Pipeline(steps=[('vectorizer', CountVectorizer()), ('nb',
MultinomialNB())])
```

```
Pipeline(steps=[('vectorizer', CountVectorizer()), ('nb',
MultinomialNB())])
```

```
emails=[
    'Sounds great! Are you home now?',
    'Will u meet ur dream partner soon? Is ur career off 2 a flyng
start? 2 find out free, txt HORO followed by ur star sign, e. g. HORO
ARIES'
]

clt.predict(emails)

array([1, 0])

clt.score(x_test,y_test)

0.9845320959010054

clt.score(x_train, y_train)

0.9938080495356038

emails=[
    'I m gonna be home soon and i dont want to talk about this stuff
anymore tonight, k? Ive cried enough today.',
    'WINNER!! As a valued network customer you have been selected to
receivea å£900 prize reward! To claim call 09061701461. Claim code
KL341. Valid 12 hours only.'
]

clt.predict(emails)

array([1, 0])

emails=[
    'Had your mobile 11 months or more? U R entitled to Update to the
latest colour mobiles with camera for Free! Call The Mobile Update Co
FREE on 08002986030',
    'SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and
send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4
info'
]

clt.predict(emails)

array([0, 0])
```

End of the Code