SPAM EMAIL DETECTION

ML Project – Phase 1

Harsha Sathish

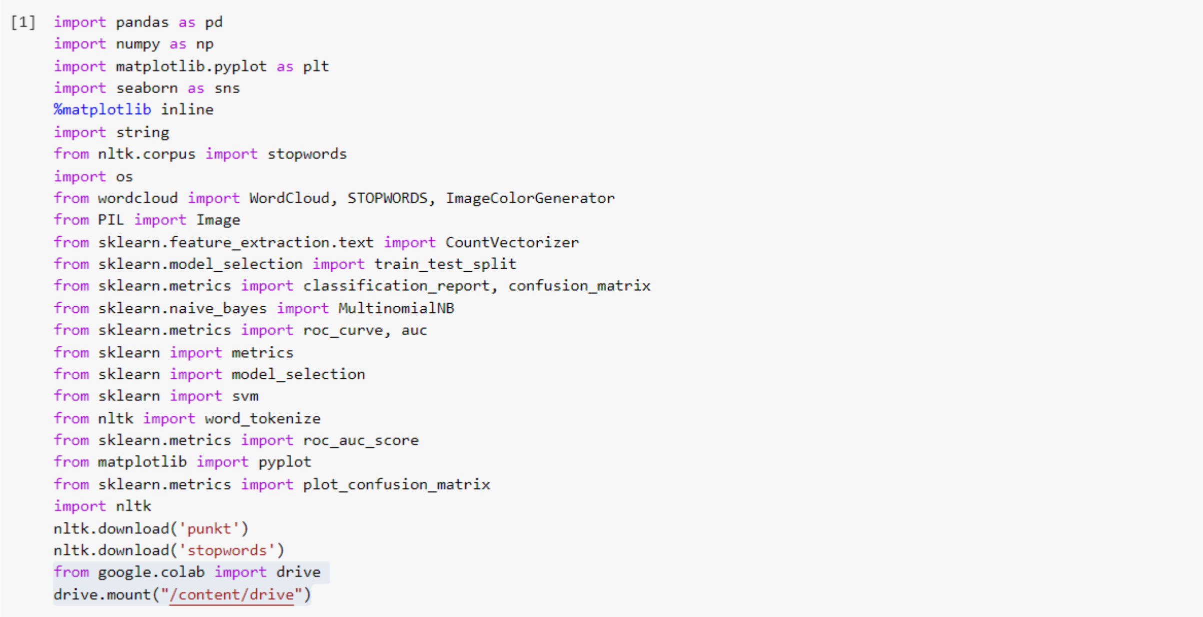
AM.EN.U4CSE19123

Introduction:

In the Spam Email Detection Project, we are provided with an unprocessed dataset having only 2 features. One of the features is the spam email contents, namely text, and the other feature, specifying whether the particular email is classified as spam or ham. Therefore, for implementing the ML algorithm, we have to pre-process the given dataset. That is, we need to remove the stopwords and punctuation from the raw data.

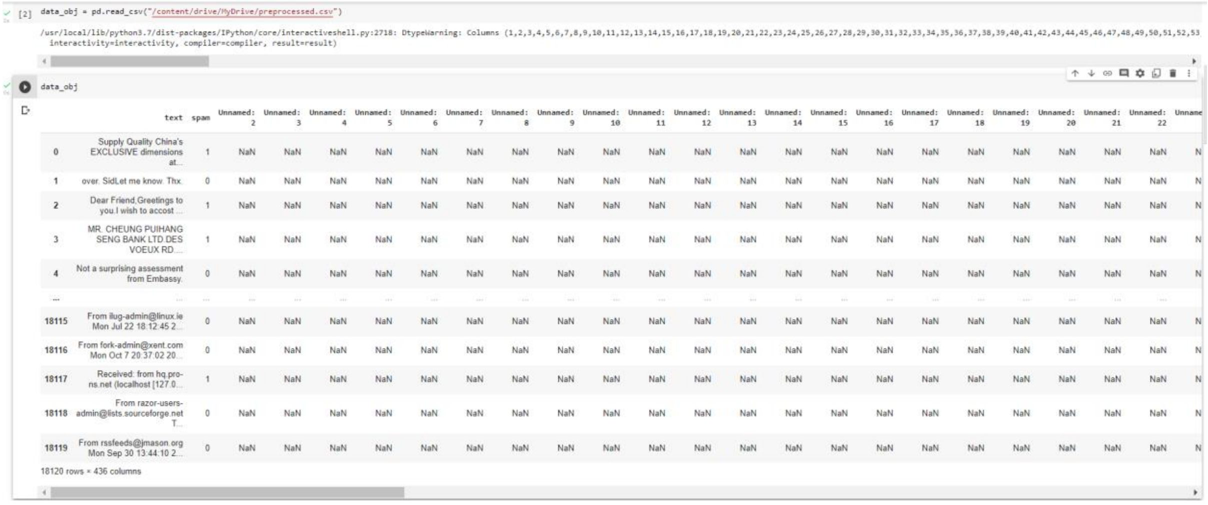
Data Pre-processing

- Import all necessary packages



```
[1] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn as sns
    %matplotlib inline
    import string
    from nltk.corpus import stopwords
    import os
    from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
    from PIL import Image
    from sklearn.feature_extraction.text import CountVectorizer
    from sklearn.model_selection import train_test_split
    from sklearn.metrics import classification_report, confusion_matrix
    from sklearn.naive_bayes import MultinomialNB
    from sklearn.metrics import roc_curve, auc
    from sklearn import metrics
    from sklearn import model_selection
    from sklearn import svm
    from nltk import word_tokenize
    from sklearn.metrics import roc_auc_score
    from matplotlib import pyplot
    from sklearn.metrics import plot_confusion_matrix
    import nltk
    nltk.download('punkt')
    nltk.download('stopwords')
    from google.colab import drive
    drive.mount("/content/drive")
```

- Pre-requisites: Load the dataset and display the contents



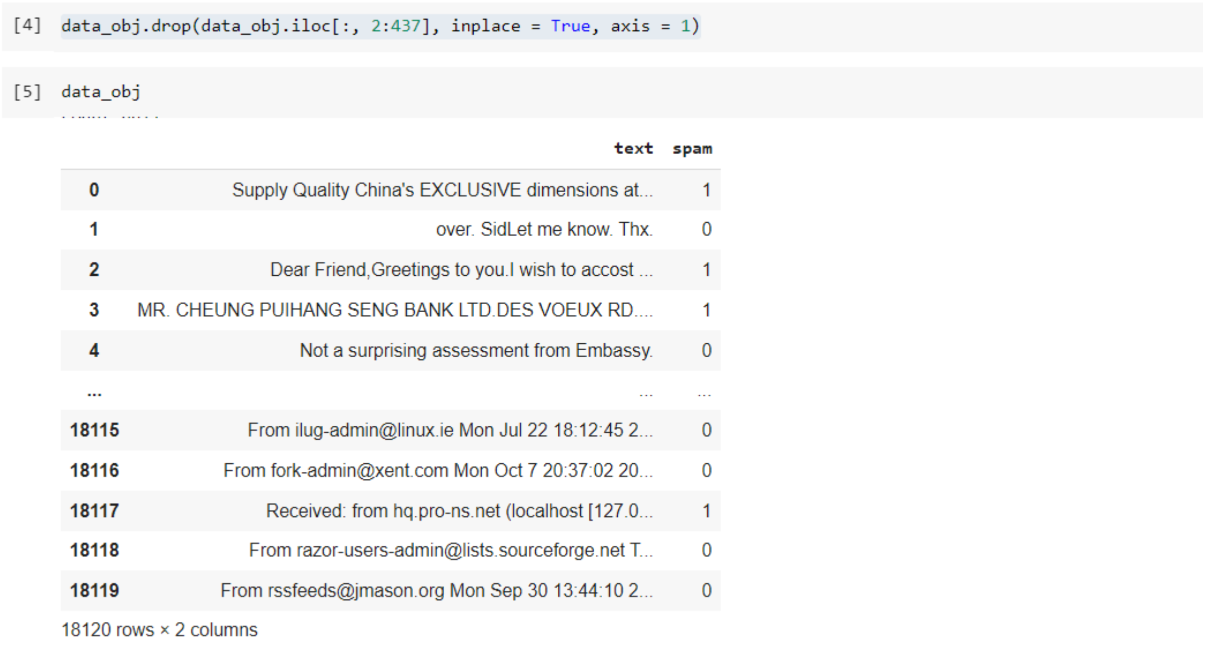
```
[2] data_obj = pd.read_csv("/content/drive/MyDrive/preprocessed.csv")

/usr/local/lib/python3.7/dist-packages/IPython/core/interactiveshell.py:2718: DtypeWarning: Columns (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,48,49,50,51,52,53
  interactivity=interactivity, compiler=compiler, result=result)
```

```
data_obj
```

	text	spam	Unnamed: 2	Unnamed: 3	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	Unnamed: 8	Unnamed: 9	Unnamed: 10	Unnamed: 11	Unnamed: 12	Unnamed: 13	Unnamed: 14	Unnamed: 15	Unnamed: 16	Unnamed: 17	Unnamed: 18	Unnamed: 19	Unnamed: 20	Unnamed: 21	Unnamed: 22	Unname
0	Supply Quality China's EXCLUSIVE dimensions at...	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
1	over. SidLet me know. Thx.	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
2	Dear Friend,Greetings to you.I wish to accost ...	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
3	MR. CHEUNG PUIHANG SENG BANK LTD.DES VOEUX RD....	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
4	Not a surprising assessment from Embassy.	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
18115	From ilug-admin@linux.ie Mon Jul 22 18:12:45 2...	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
18116	From fork-admin@xent.com Mon Oct 7 20:37:02 20...	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
18117	Received: from hq.pro-ns.net (localhost [127.0...	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
18118	From razor-users-admin@lists.sourceforge.net T...	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
18119	From rssfeeds@jmason.org Mon Sep 30 13:44:10 2...	0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N

18120 rows × 436 columns

- Remove the unwanted rows and display the dataset

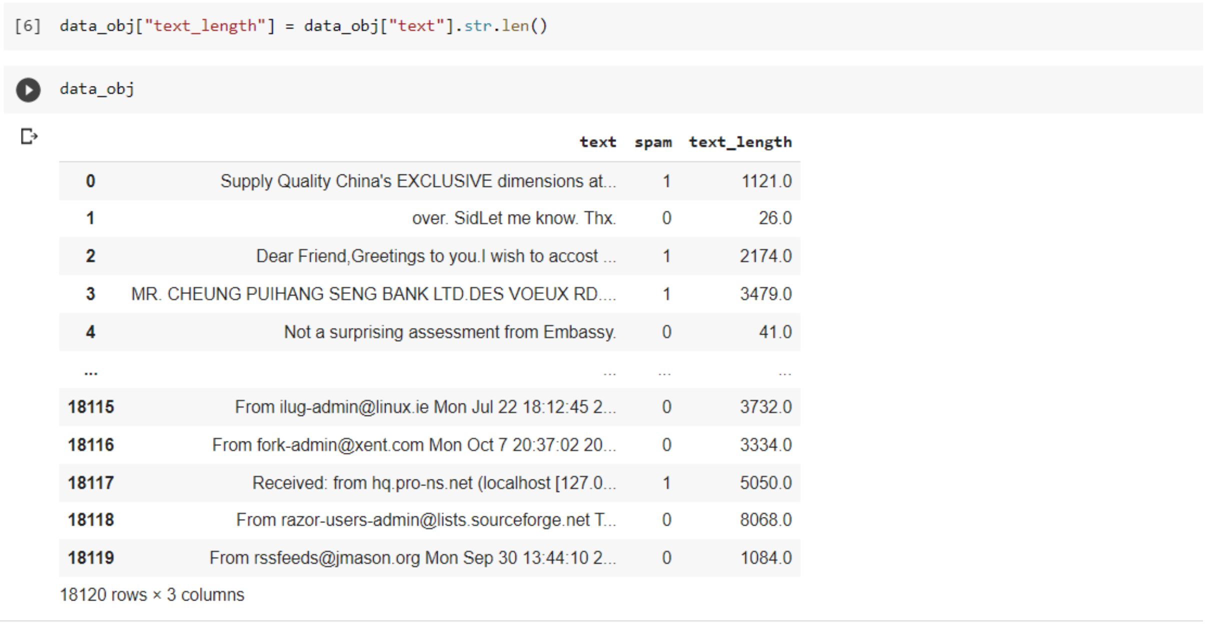


```
[4]  data_obj.drop(data_obj.iloc[:, 2:437], inplace = True, axis = 1)
```

```
[5]  data_obj
```

```
                                                  text  spam
0        Supply Quality China's EXCLUSIVE dimensions at...     1
1                          over. SidLet me know. Thx.     0
2           Dear Friend,Greetings to you.I wish to accost ...     1
3     MR. CHEUNG PUIHANG SENG BANK LTD.DES VOEUX RD....     1
4                Not a surprising assessment from Embassy.     0
...                                                    ...   ...
18115      From ilug-admin@linux.ie Mon Jul 22 18:12:45 2...     0
18116     From fork-admin@xent.com Mon Oct 7 20:37:02 20...     0
18117       Received: from hq.pro-ns.net (localhost [127.0...     1
18118     From razor-users-admin@lists.sourceforge.net T...     0
18119    From rssfeeds@jmason.org Mon Sep 30 13:44:10 2...     0
18120 rows × 2 columns
```

- Add the text length column in each record

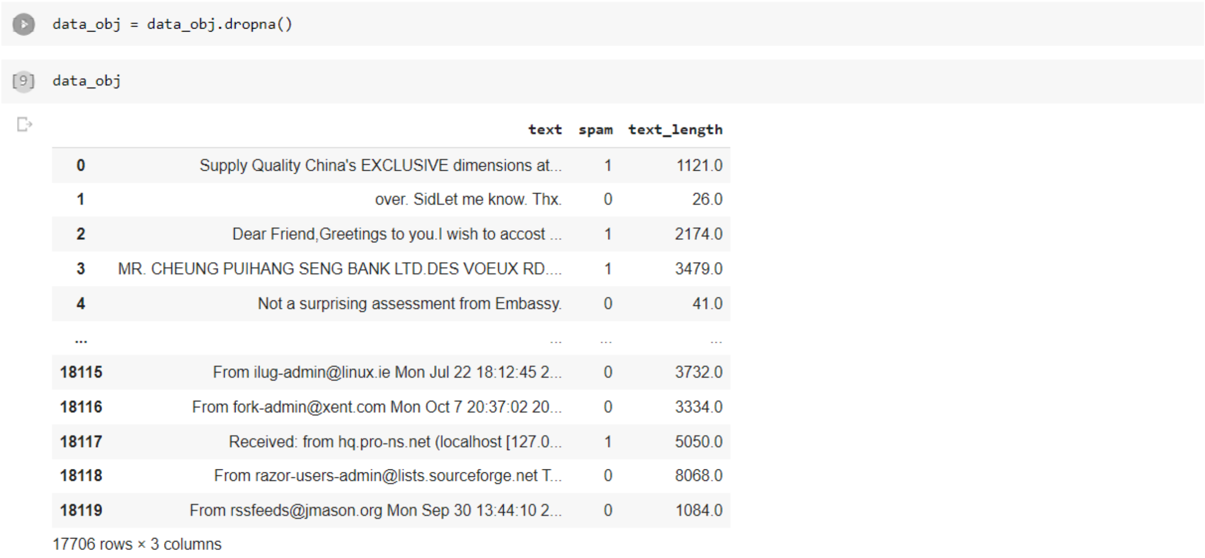


```
[6]  data_obj["text_length"] = data_obj["text"].str.len()
```

```
     data_obj
```

```
                                                  text  spam  text_length
0        Supply Quality China's EXCLUSIVE dimensions at...     1       1121.0
1                          over. SidLet me know. Thx.     0         26.0
2           Dear Friend,Greetings to you.I wish to accost ...     1       2174.0
3     MR. CHEUNG PUIHANG SENG BANK LTD.DES VOEUX RD....     1       3479.0
4                Not a surprising assessment from Embassy.     0         41.0
...                                                    ...   ...          ...
18115      From ilug-admin@linux.ie Mon Jul 22 18:12:45 2...     0       3732.0
18116     From fork-admin@xent.com Mon Oct 7 20:37:02 20...     0       3334.0
18117       Received: from hq.pro-ns.net (localhost [127.0...     1       5050.0
18118     From razor-users-admin@lists.sourceforge.net T...     0       8068.0
18119    From rssfeeds@jmason.org Mon Sep 30 13:44:10 2...     0       1084.0
18120 rows × 3 columns
```

- Search for columns with null values, and delete them



```
data_obj = data_obj.dropna()
```

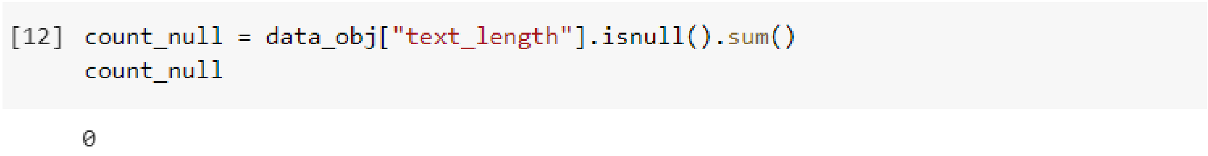
```
[9] data_obj
```

	text	spam	text_length
0	Supply Quality China's EXCLUSIVE dimensions at...	1	1121.0
1	over. SidLet me know. Thx.	0	26.0
2	Dear Friend,Greetings to you.I wish to accost ...	1	2174.0
3	MR. CHEUNG PUIHANG SENG BANK LTD.DES VOEUX RD....	1	3479.0
4	Not a surprising assessment from Embassy.	0	41.0
...	...	...	...
18115	From ilug-admin@linux.ie Mon Jul 22 18:12:45 2...	0	3732.0
18116	From fork-admin@xent.com Mon Oct 7 20:37:02 20...	0	3334.0
18117	Received: from hq.pro-ns.net (localhost [127.0...	1	5050.0
18118	From razor-users-admin@lists.sourceforge.net T...	0	8068.0
18119	From rssfeeds@jmason.org Mon Sep 30 13:44:10 2...	0	1084.0

17706 rows × 3 columns

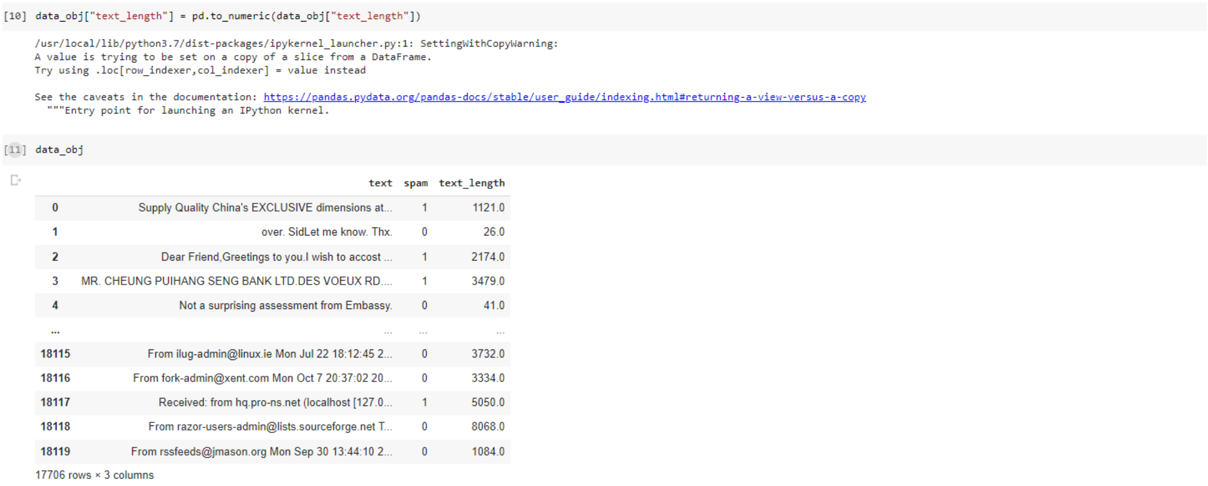
The **dropna()** returns the dataframe with columns having null values removed. The returned dataframe is stored into data_obj.

- Re-check if any columns with null values are left after deletion



```
[12] count_null = data_obj["text_length"].isnull().sum()
     count_null

     0
```

- Convert datatype of text_length to numeric and display the dataset



```
[10] data_obj["text_length"] = pd.to_numeric(data_obj["text_length"])

     /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py: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
       """Entry point for launching an IPython kernel.
```

```
[11] data_obj
```

	text	spam	text_length
0	Supply Quality China's EXCLUSIVE dimensions at...	1	1121.0
1	over. SidLet me know. Thx.	0	26.0
2	Dear Friend,Greetings to you.I wish to accost ...	1	2174.0
3	MR. CHEUNG PUIHANG SENG BANK LTD.DES VOEUX RD....	1	3479.0
4	Not a surprising assessment from Embassy.	0	41.0
...	...	...	...
18115	From ilug-admin@linux.ie Mon Jul 22 18:12:45 2...	0	3732.0
18116	From fork-admin@xent.com Mon Oct 7 20:37:02 20...	0	3334.0
18117	Received: from hq.pro-ns.net (localhost [127.0...	1	5050.0
18118	From razor-users-admin@lists.sourceforge.net T...	0	8068.0
18119	From rssfeeds@jmason.org Mon Sep 30 13:44:10 2...	0	1084.0

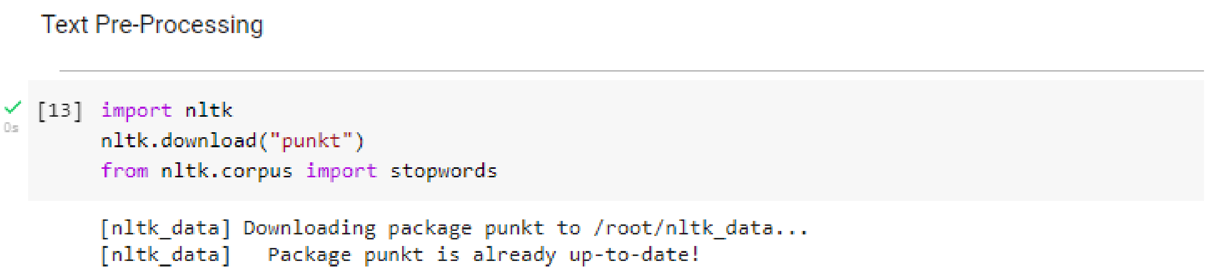
17706 rows × 3 columns

Word Tokenization

- ❖ Convert all characters in text into LowerCase.
- ❖ Remove Special Characters
- ❖ Remove all stopwords in English Language using NLTK
- ❖ Remove Hyperlinks
- ❖ Remove words with similar meaning

Tokenization of the data is the splitting of text into smaller chunks, which are called tokens.

- Import NLTK Library

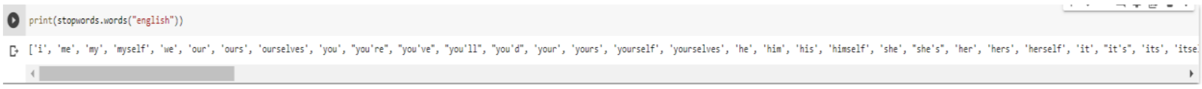


Text Pre-Processing

```
[13] import nltk
     nltk.download("punkt")
     from nltk.corpus import stopwords

     [nltk_data] Downloading package punkt to /root/nltk_data...
     [nltk_data]   Package punkt is already up-to-date!
```

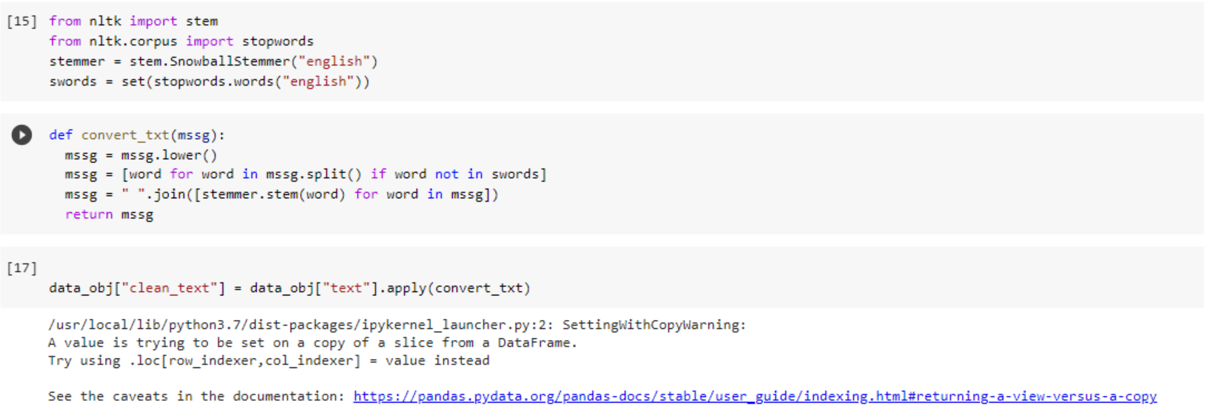
nltk or Natural Language Toolkit is a suite of libraries and programs in Natural Language Processing (NLP) for English in Python.

- List the stop words, which may be removed from the dirty text



```
print(stopwords.words("english"))

['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itse
```

- Convert the text to lowercase and remove the stopwords



```
[15] from nltk import stem
     from nltk.corpus import stopwords
     stemmer = stem.SnowballStemmer("english")
     swords = set(stopwords.words("english"))
```

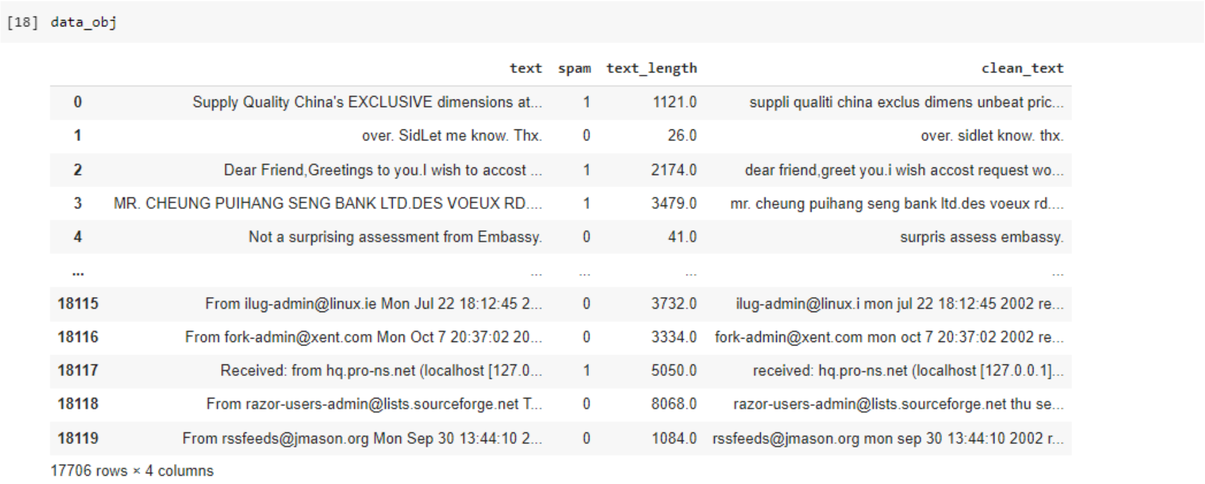
```
def convert_txt(mssg):
  mssg = mssg.lower()
  mssg = [word for word in mssg.split() if word not in swords]
  mssg = " ".join([stemmer.stem(word) for word in mssg])
  return mssg
```

```
[17]
     data_obj["clean_text"] = data_obj["text"].apply(convert_txt)

     /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: 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
```

In the convert_txt(), the text is being converted into lowercase. Then, it checks if the particular word is present in swords (set of stopwords in English). If the word is present in the set, it is removed. Now, the text is subjected to stemming.

Stemming is the process of producing morphological variants of a root word. It removed redundancy as most of the time the word stem and their derived words mean the same.

- Display the processed dataset



```
[18] data_obj
```

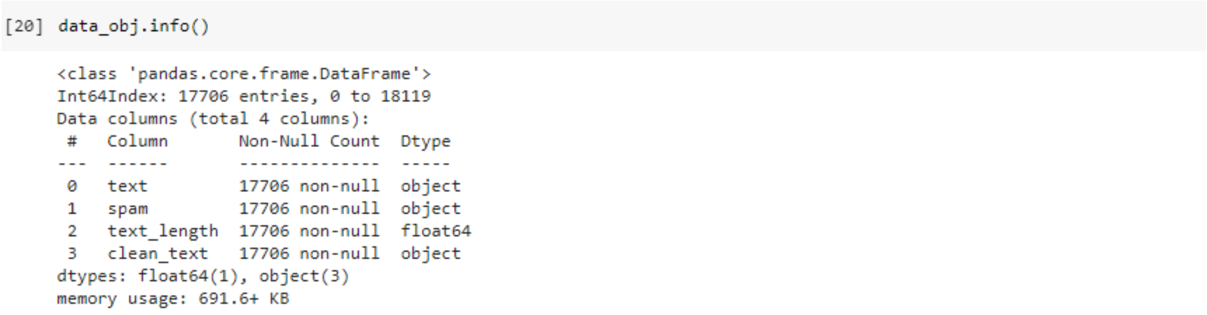
	text	spam	text_length	clean_text
0	Supply Quality China's EXCLUSIVE dimensions at...	1	1121.0	suppli qualiti china exclus dimens unbeat pric...
1	over. SidLet me know. Thx.	0	26.0	over. sidlet know. thx.
2	Dear Friend,Greetings to you.I wish to accost ...	1	2174.0	dear friend,greet you.i wish accost request wo...
3	MR. CHEUNG PUIHANG SENG BANK LTD.DES VOEUX RD....	1	3479.0	mr. cheung puihang seng bank ltd.des voeux rd....
4	Not a surprising assessment from Embassy.	0	41.0	surpris assess embassy.
...	...	...	...	...
18115	From ilug-admin@linux.ie Mon Jul 22 18:12:45 2...	0	3732.0	ilug-admin@linux.i mon jul 22 18:12:45 2002 re...
18116	From fork-admin@xent.com Mon Oct 7 20:37:02 20...	0	3334.0	fork-admin@xent.com mon oct 7 20:37:02 2002 re...
18117	Received: from hq.pro-ns.net (localhost [127.0...	1	5050.0	received: hq.pro-ns.net (localhost [127.0.0.1]...
18118	From razor-users-admin@lists.sourceforge.net T...	0	8068.0	razor-users-admin@lists.sourceforge.net thu se...
18119	From rssfeeds@jmason.org Mon Sep 30 13:44:10 2...	0	1084.0	rssfeeds@jmason.org mon sep 30 13:44:10 2002 r...

17706 rows × 4 columns

Data Summarization

Using various methods in Pandas framework, a descriptive analysis is possible to describe the basic features of the dataset and obtain a brief summary of the data.

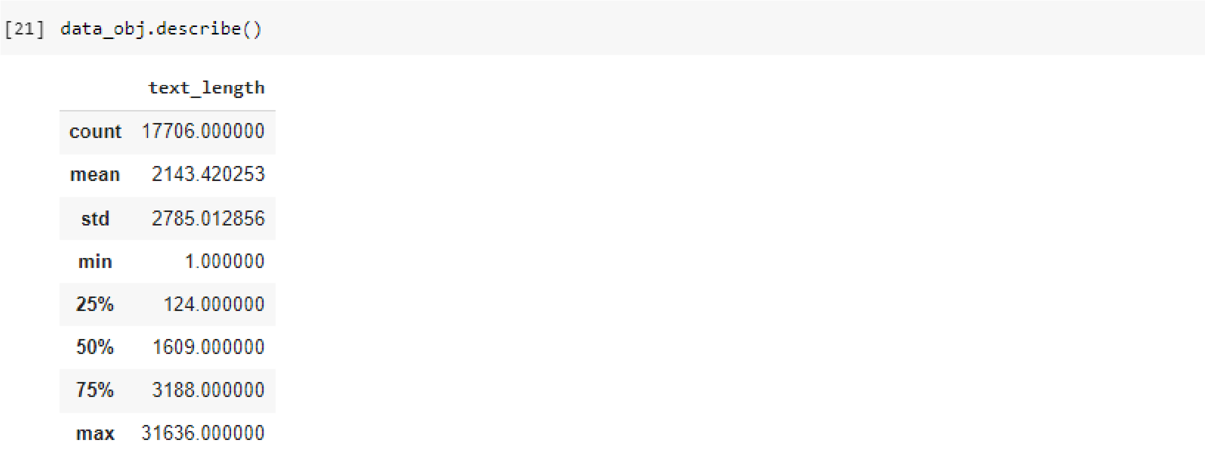
- Using info(), we get a quick overview of the dataset.



```
[20] data_obj.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 17706 entries, 0 to 18119
Data columns (total 4 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   text         17706 non-null  object 
 1   spam         17706 non-null  object 
 2   text_length  17706 non-null  float64
 3   clean_text   17706 non-null  object 
dtypes: float64(1), object(3)
memory usage: 691.6+ KB
```

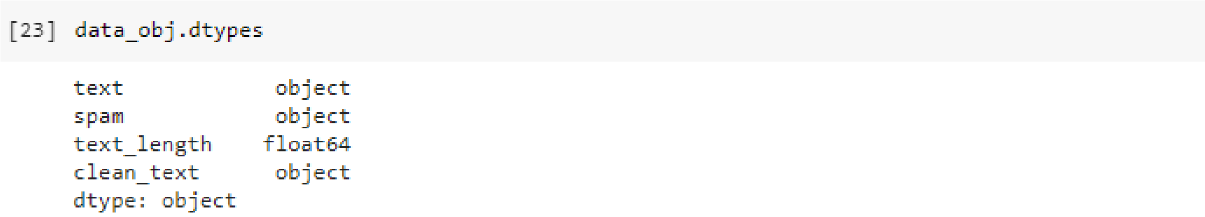
- Using Pandas describe(), we can view the statistical data such as percentile, mean, std etc. of a data frame.



```
[21] data_obj.describe()
```

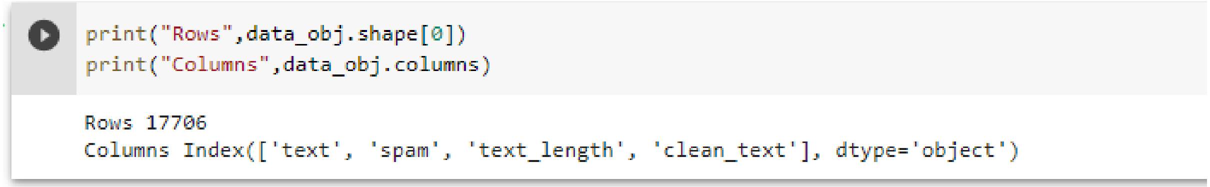
	text_length
count	17706.000000
mean	2143.420253
std	2785.012856
min	1.000000
25%	124.000000
50%	1609.000000
75%	3188.000000
max	31636.000000

- Dataobj_dtypes displays the datatypes in the DataFrame.



```
[23] data_obj.dtypes

     text             object
     spam             object
     text_length     float64
     clean_text       object
     dtype: object
```

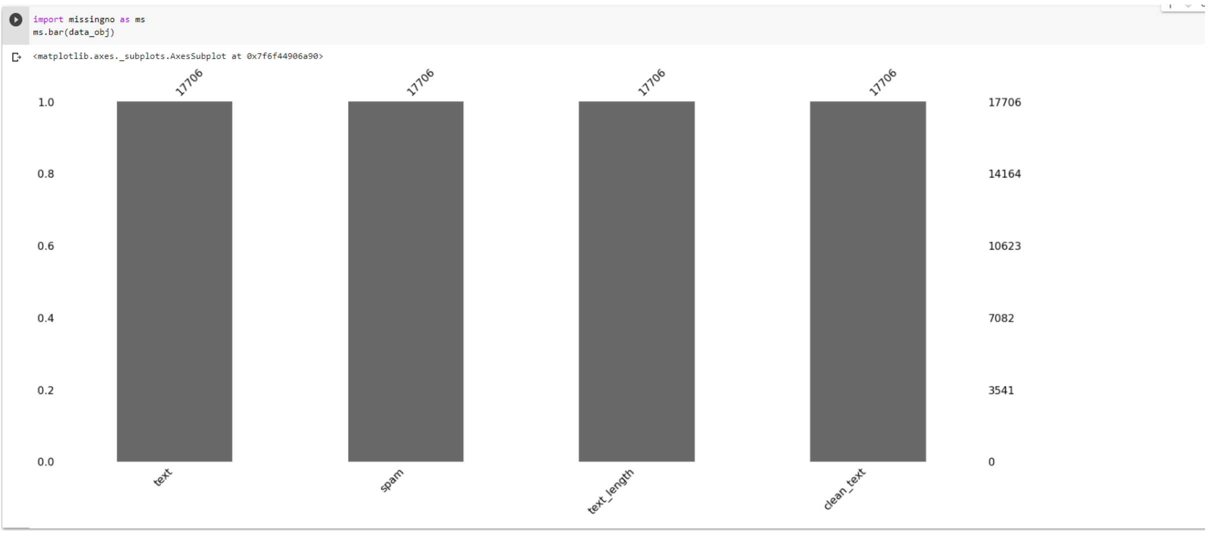
- Calculate the number of rows and columns in the csv file



```
print("Rows",data_obj.shape[0])
print("Columns",data_obj.columns)

Rows 17706
Columns Index(['text', 'spam', 'text_length', 'clean_text'], dtype='object')
```

Data Visualization

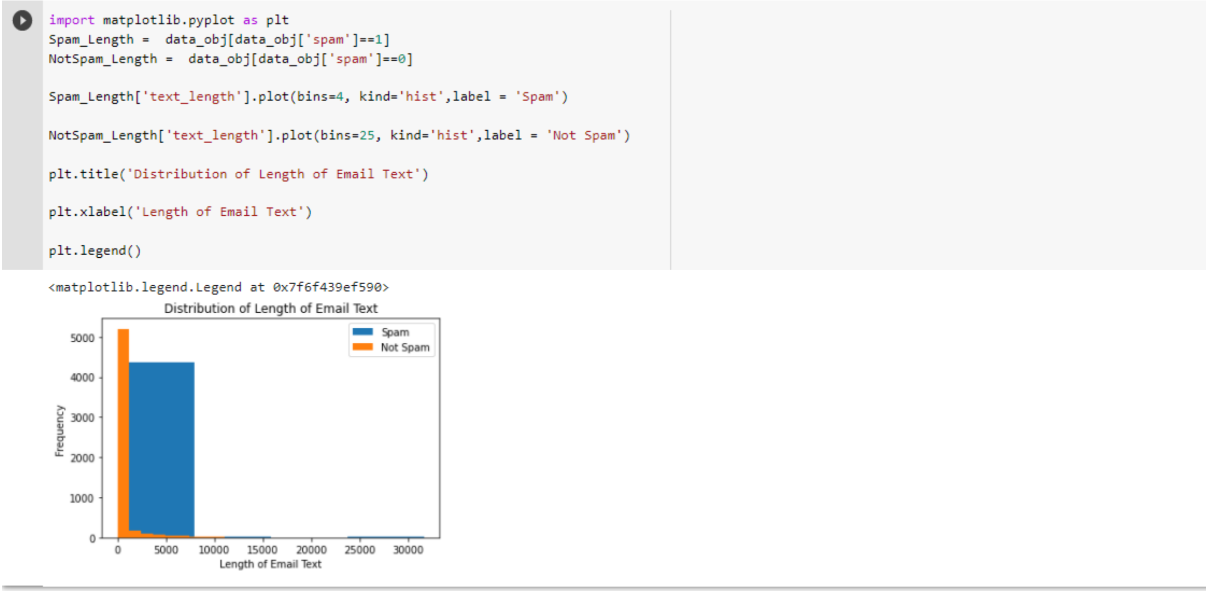
- Using Bar Graph, we are able to see where the missing values are located in each column and the correlation between missing values of different columns.



```
import missingno as ms
ms.bar(data_obj)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f6f44906a90>
```

- Using matplotlib, we can represent comparisons between discrete categories. In the graph below, x-axis represents the length of the email text and the y-axis represents the frequency corresponding to those categories.



```
import matplotlib.pyplot as plt
Spam_Length =  data_obj[data_obj['spam']==1]
NotSpam_Length =  data_obj[data_obj['spam']==0]

Spam_Length['text_length'].plot(bins=4, kind='hist',label = 'Spam')

NotSpam_Length['text_length'].plot(bins=25, kind='hist',label = 'Not Spam')

plt.title('Distribution of Length of Email Text')

plt.xlabel('Length of Email Text')

plt.legend()
```

```
<matplotlib.legend.Legend at 0x7f6f439ef590>
```

Data Interpretation:

- Dataset before processing: Drive Link
- Dataset after processing : Drive Link