Importing required libraries

In [1]:
```
import pandas as pd
```

Load data

In [2]:
```
raw_mails = pd.read_csv('spam.csv', encoding = 'latin')
print(raw_mails)
```

```
        v1                                                 v2 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]
```

Preprocess Data

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

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

In [4]:
```
raw_mails.describe()
```

Out[4]:

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
count	5572	5572	50	12	6
unique	2	5169	43	10	5
top	ham	Sorry, I'll call later	bt not his girlfrnd... G o o d n i g h t . . .@"	MK17 92H. 450Ppw 16"	GNT:-)"
freq	4825	30	3	2	2

In [5]:
```
raw_mails.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   v1          5572 non-null   object
 1   v2          5572 non-null   object
 2   Unnamed: 2  50 non-null     object
 3   Unnamed: 3  12 non-null     object
 4   Unnamed: 4  6 non-null      object
dtypes: object(5)
memory usage: 217.8+ KB
```

In [6]:
```
raw_mails.replace({'v1' : {'spam' : 0, 'ham' : 1}}, inplace = True)
```

In [7]:
```
x = raw_mails['v2']
y = raw_mails['v1']
x
```

Out[7]:
```
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: v2, Length: 5572, dtype: object
```

In [8]:
```
print(y)
```

```
0       1
1       1
2       0
3       1
4       1
       ..
5567    0
5568    1
5569    1
5570    1
5571    1
Name: v1, Length: 5572, dtype: int64
```

Model Training

In [9]:
```
import matplotlib.pyplot as plt
```

In [10]:
```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_stat
```

In [11]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
feature_extraction = TfidfVectorizer()
y_train = y_train.astype('int')
y_test = y_test.astype('int')
x_train_features = feature_extraction.fit_transform(x_train)
x_test_features = feature_extraction.transform(x_test)
```

In [12]:
```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(x_train_features, y_train)
```

Out[12]:
```
▾ LogisticRegression

LogisticRegression()
```

In [13]:
```
y_predict = lr.predict(x_test_features)
y_predict
```

Out[13]:
```
array([1, 0, 1, ..., 1, 1, 1])
```

In [14]:
```
from sklearn.metrics import accuracy_score
accuracy_score_data = accuracy_score(y_test, y_predict)
print("Accuracy score = ", accuracy_score_data)
```

```
Accuracy score =  0.9757847533632287
```