In [44]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
```

In [45]:

```
df=pd.read_csv("C:/Users/Jahnavi/Downloads/Mail/spam.csv",encoding = "ISO-8859-1")
```

In [46]:

```
df
```

Out[46]:

	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 [47]:

```
df.head()
```

Out[47]:

	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

In [48]:

```
df.tail()
```

Out[48]:

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
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

In [49]:

```
df.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 [50]:

```
df.isnull().sum()
```

Out[50]:

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

In [54]:

```
df.drop_duplicates(inplace=True)
print(df.shape)
```

```
(5169, 5)
```

In [55]:

```
df1.duplicated().sum()
```

Out[55]:

```
0
```

In [56]:

```
df.describe()
```

Out[56]:

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
count	5169	5169	43	10	5
unique	2	5169	43	10	5
top	ham	Go until jurong point, crazy.. Available only ...	PO Box 5249	MK17 92H. 450Ppw 16"	just Keep-in-touch\" gdeve.."
freq	4516	1	1	1	1

In [57]:

```
df1 = df.drop(["Unnamed: 2","Unnamed: 3","Unnamed: 4"], axis=1)
```

In [58]:

```
df1.head()
```

Out[58]:

	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 [59]:

```
df1.rename(columns = {"v1" : "Category", "v2":"Message"},inplace = True)
df1.head()
```

Out[59]:

	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...

In [60]:

```
from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
df1['Category'] = encoder.fit_transform(df1['Category'])
```

In [61]:

```
X =df1["Message"]
y =df1["Category"]
```

In [62]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train,y_test = train_test_split(X,y,test_size = 0.20, random_state = 0)
```

In [63]:

```
from sklearn.feature_extraction.text import CountVectorizer
cv = CountVectorizer()
X_train_count = cv.fit_transform(X_train.values)
X_train_count.toarray()
```

Out[63]:

```
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]], dtype=int64)
```

In [64]:

```
from sklearn.naive_bayes import MultinomialNB
model = MultinomialNB()
model.fit(X_train_count,y_train)
```

Out[64]:

```
MultinomialNB()
```

In [65]:

```
from sklearn.metrics import confusion_matrix , recall_score , precision_score
from sklearn.metrics import accuracy_score
```

In [69]:

```
ham = ['Same. You are soo right']
ham_count = cv.transform(ham)
y_pred = model.predict(ham_count)
y_pred
```

Out[69]:

```
array([0])
```

In [70]:

```
model.score(X_train_count,y_train)
```

Out[70]:

```
0.9929866989117292
```

In [71]:

```
X_test_count = cv.transform(X_test)
model.score(X_test_count,y_test)
```

Out[71]:

```
0.9816247582205029
```

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js