In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [3]:
```
dataset=pd.read_csv('spam.csv',encoding='latin-1')
```

In [4]:
```
dataset.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 [5]:
```
dataset.head()
```

Out[5]:

	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 [ ]:
```

```

In [6]:
```
dataset.drop(['Unnamed: 2','Unnamed: 3','Unnamed: 4'],axis=1,inplace=True)
```

In [7]:
```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   v1      5572 non-null   object
 1   v2      5572 non-null   object
dtypes: object(2)
memory usage: 87.2+ KB
```

In [8]:
```
dataset.head()
```

Out[8]:

	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 [9]:
```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
dataset['v1']=le.fit_transform(dataset['v1'])
```

In [10]:
```
dataset.head()
```

Out[10]:

	v1	v2
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

In [11]:
```
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
Corpus=[]
for i in range(0,5572):
  review=re.sub('[^a-zA-Z]',' ',dataset['v2'][i])
  review=review.lower()
  review=review.split()
  ps=PorterStemmer()
  all_stopwords=stopwords.words('english')
  all_stopwords.remove('not')
  review=[ps.stem(word) for word in review if not word in set(all_stopwords)]
  review=' '.join(review)
  Corpus.append(review)
dataset['v2']=Corpus
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Sanjana.L\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [12]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
Vectorizer=TfidfVectorizer()
X=dataset['v2']
y=dataset['v1']
X=Vectorizer.fit_transform(X).toarray()
```

In [13]:
```
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_state=0)
```

In [14]:
```
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import confusion_matrix, accuracy_score
classifier = MultinomialNB()
classifier.fit(X_train, y_train)
y_pred1= classifier.predict(X_test)
cm = confusion_matrix(y_test, y_pred1)
print(cm)
print("accuracy:"+str(accuracy_score(y_test,y_pred1)))
```

```
[[949   0]
 [ 48 118]]
accuracy:0.9569506726457
```

In [15]:
```
from sklearn.metrics import confusion_matrix, accuracy_score
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0)
classifier.fit(X_train, y_train)
y_pred2=classifier.predict(X_test)
cm = confusion_matrix(y_test, y_pred2)
print(cm)
print("accuracy:"+str(accuracy_score(y_test,y_pred2)))
```

```
[[948   1]
 [ 46 120]]
accuracy:0.9578475336322287
```

In [16]:
```
from sklearn.svm import SVC
classifier = SVC(kernel = 'linear', random_state = 0)
classifier.fit(X_train, y_train)
y_pred3 = classifier.predict(X_test)
cm = confusion_matrix(y_test,y_pred3)
print(cm)
print("accuracy:"+str(accuracy_score(y_test,y_pred3)))
```

```
[[948   1]
 [ 24 142]]
accuracy:0.9775784753363229
```

In [17]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [18]:
```
dataset=pd.read_csv('spam.csv',encoding='latin-1')
```

In [19]:
```
dataset.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 [20]:
```
dataset.head()
```

Out[20]:

	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 [22]:
```
dataset.drop(['Unnamed: 2','Unnamed: 3','Unnamed: 4'],axis=1,inplace=True)
```

In [23]:
```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5572 entries, 0 to 5571
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   v1      5572 non-null   object
 1   v2      5572 non-null   object
dtypes: object(2)
memory usage: 87.2+ KB
```

In [24]:
```
dataset.head()
```

Out[24]:

	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 [25]:
```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
dataset['v1']=le.fit_transform(dataset['v1'])
```

In [26]:
```
dataset.head()
```

Out[26]:

	v1	v2
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

In [27]:
```
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
Corpus=[]
for i in range(0,5572):
  review=re.sub('[^a-zA-Z]',' ',dataset['v2'][i])
  review=review.lower()
  review=review.split()
  ps=PorterStemmer()
  all_stopwords=stopwords.words('english')
  all_stopwords.remove('not')
  review=[ps.stem(word) for word in review if not word in set(all_stopwords)]
  review=' '.join(review)
  Corpus.append(review)
dataset['v2']=Corpus
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Sanjana.L\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [28]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
Vectorizer=TfidfVectorizer()
X=dataset['v2']
y=dataset['v1']
X=Vectorizer.fit_transform(X).toarray()
```

In [29]:
```
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_state=0)
```

In [30]:
```
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import confusion_matrix, accuracy_score
classifier = MultinomialNB()
classifier.fit(X_train, y_train)
y_pred1= classifier.predict(X_test)
cm = confusion_matrix(y_test, y_pred1)
print(cm)
print("accuracy:"+str(accuracy_score(y_test,y_pred1)))
```

```
[[949   0]
 [ 48 118]]
accuracy:0.9569506726457
```

In [31]:
```
from sklearn.metrics import confusion_matrix, accuracy_score
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0)
classifier.fit(X_train, y_train)
y_pred2=classifier.predict(X_test)
cm = confusion_matrix(y_test, y_pred2)
print(cm)
print("accuracy:"+str(accuracy_score(y_test,y_pred2)))
```

```
[[948   1]
 [ 46 120]]
accuracy:0.9578475336322287
```

In [32]:
```
from sklearn.svm import SVC
classifier = SVC(kernel = 'linear', random_state = 0)
classifier.fit(X_train, y_train)
y_pred3 = classifier.predict(X_test)
cm = confusion_matrix(y_test,y_pred3)
print(cm)
print("accuracy:"+str(accuracy_score(y_test,y_pred3)))
```

```
[[948   1]
 [ 24 142]]
accuracy:0.9775784753363229
```

In [ ]: