In [1]:
```
pip install nltk
```

```
Requirement already satisfied: nltk in c:\users\computer\anaconda3\lib\site-packages (3.5)Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: regex in c:\users\computer\anaconda3\lib\site-packages (from nltk) (2020.10.15)
Requirement already satisfied: click in c:\users\computer\anaconda3\lib\site-packages (from nltk) (7.1.2)
Requirement already satisfied: joblib in c:\users\computer\anaconda3\lib\site-packages (from nltk) (0.17.0)
Requirement already satisfied: tqdm in c:\users\computer\anaconda3\lib\site-packages (from nltk) (4.50.2)
```

In [2]:
```
import nltk
```

In [3]:
```
nltk.download('wordnet')
```

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

Out[3]:
```
True
```

loading dataset

In [4]:
```
import pandas as pd
```

In [9]:
```
dt=pd.read_csv("spam.csv",encoding='Windows-1252')
```

In [6]:
```
import chardet
with open("spam.csv",'rb') as rawdata:
    result=chardet.detect(rawdata.read(100000))
result
```

Out[6]:
```
{'encoding': 'Windows-1252', 'confidence': 0.727032249892918, 'language': ''}
```

In [11]:
```
dt.head(10)
```

Out[11]:

	type	text
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...
5	spam	FreeMsg Hey there darling it's been 3 week's n...
6	ham	Even my brother is not like to speak with me. ....
7	ham	As per your request 'Melle Melle (Oru Minnamin...
8	spam	WINNER!! As a valued network customer you have...
9	spam	Had your mobile 11 months or more? U R entitle...

In [12]:
```
dt['spam'] = dt['type'].map( {'spam': 1, 'ham': 0} ).astype(int)
dt.head(5)
```

Out[12]:

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

In [13]:
```
print("Columns in the given data")
for col in dt.columns:
    print(col)
```

```
Columns in the given data
type
text
spam
```

In [15]:
```
t=len(dt['type'])
print("NO OF ROWS IN REVIEW COLUMN:",t)
t=len(dt['text'])
print("NO OF ROWS IN liked COLUMN:",t)
```

```
NO OF ROWS IN REVIEW COLUMN: 116
NO OF ROWS IN liked COLUMN: 116
```

Tokenization

In [16]:
```
dt['text'][4]
```

Out[16]:
```
"Nah I don't think he goes to usf, he lives around here though"
```

In [17]:
```
def tokenizer(text):
    return text.split()
```

In [18]:
```
dt['text']=dt['text'].apply(tokenizer)
```

In [19]:
```
dt['text'][4]
```

Out[19]:
```
['Nah',
 'I',
 "don't",
 'think',
 'he',
 'goes',
 'to',
 'usf,',
 'he',
 'lives',
 'around',
 'here',
 'though']
```

STEMMING

In [20]:
```
from nltk.stem.snowball import SnowballStemmer
porter = SnowballStemmer("english", ignore_stopwords=False)
```

In [22]:
```
def stem_it(text):
    return [porter.stem(word) for word in text]
```

In [23]:
```
dt['text']=dt['text'].apply(stem_it)
dt['text'][4]
```

Out[23]:
```
['nah',
 'i',
 "don't",
 'think',
 'he',
 'goe',
 'to',
 'usf,',
 'he',
 'live',
 'around',
 'here',
 'though']
```

LEMMITIZATION

In [24]:
```
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
```

In [26]:
```
def lemmit_it(text):
    return [lemmatizer.lemmatize(word, pos ="a") for word in text]
```

In [ ]:
```

```

In [27]:
```
from nltk.corpus import stopwords
stop_words=stopwords.words('english')
```

In [28]:
```
nltk.download('stopwords')
```

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

Out[28]:
```
True
```

In [30]:
```
def stop_it(text):
    review = [word for word in text if not word in stop_words ]
    return review
```

In [31]:
```
dt['text']=dt['text'].apply(stop_it)
```

In [32]:
```
dt['text'][4]
```

Out[32]:
```
['nah', 'think', 'goe', 'usf,', 'live', 'around', 'though']
```

In [33]:
```
dt.head()
```

Out[33]:

	type	text	spam
0	ham	[go, jurong, point,, crazy.., avail, onli, bug...	0
1	ham	[ok, lar..., joke, wif, u, oni...]	0
2	spam	[free, entri, 2, wkli, comp, win, fa, cup, fin...	1
3	ham	[u, dun, say, earli, hor..., u, c, alreadi, sa...	0
4	ham	[nah, think, goe, usf,, live, around, though]	0

In [34]:
```
dt['text']=dt['text'].apply(' '.join)
```

In [35]:
```
dt.head()
```

Out[35]:

	type	text	spam
0	ham	go jurong point, crazy.. avail onli bugi n gre...	0
1	ham	ok lar... joke wif u oni...	0
2	spam	free entri 2 wkli comp win fa cup final tkts 2...	1
3	ham	u dun say earli hor... u c alreadi say...	0
4	ham	nah think goe usf, live around though	0

Transform Text Data into TDF /TF-IDF Vectors

In [36]:
```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf=TfidfVectorizer()
y=dt.spam.values
x=tfidf.fit_transform(dt['text'])
```

In [37]:
```
from sklearn.model_selection import train_test_split
x_train,x_text,y_train,y_text=train_test_split(x,y,random_state=1,test_size=0.2,shuffle=False)
```

In [38]:
```
from sklearn.linear_model import LogisticRegression
clf=LogisticRegression()
clf.fit(x_train,y_train)
y_pred=clf.predict(x_text)
```

In [39]:
```
from sklearn.metrics import accuracy_score
acc_log = accuracy_score(y_pred, y_text)*100
print("accuracy:",acc_log)
```

```
accuracy: 87.5
```