

In [11]:

## 1 Upload dataset

In [12]:

```
!unzip '/content/archive.zip'
```

Archive: /content/archive.zip  
replace spam.csv? [y]es, [n]o, [A]ll, [N]one, [r]ename:

## 2 Import the necessary libraries

In [14]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from keras.models import Model
from keras.layers import LSTM, Activation, Dense, Dropout, Input, Embedding
from keras.optimizers import RMSprop
from keras.preprocessing.text import Tokenizer
from keras.preprocessing import sequence
from keras.utils import to_categorical
from keras.callbacks import EarlyStopping
%matplotlib inline
```

## 3 Load the data into Pandas dataframe

In [ ]:

```
df = pd.read_csv('../input/spam.csv', delimiter=',', encoding='latin-1')
df.head()
```

## 4.Read dataset and data preprocessing

In [22]:

```
import chardet
```

In [24]:

```
with open('/content/spam.csv', 'rb') as f:
    enc = chardet.detect(f.read())
```

In [25]:

```
df=pd.read_csv('/content/spam.csv', encoding = enc['encoding'])
df.head()
```

Out[25]:

	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

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN

In [ ]:

Preprocessing

In [ ]:

In [27]:

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

Out[27]:

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

In [ ]:

In [28]:

```
df.info()  
df
```

<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

Out[28]:

	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 x 5 columns

In [6]:

In [29]:

```
df.drop(['Unnamed: 2', 'Unnamed: 3', 'Unnamed: 4'],axis=1,inplace=True)
df.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 [7]:

In [30]:

```
X = df.v2
Y = df.v1
le = LabelEncoder()
Y = le.fit_transform(Y)
Y = Y.reshape(-1,1)
```

In [8]:

In [31]:

```
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,test_size=0.15)
```

## 5 Understand the distribution better.

In [35]:

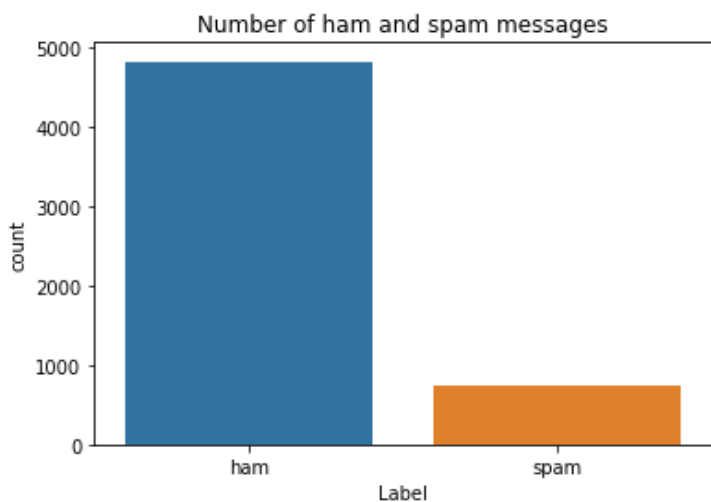
```
sns.countplot(df.v1)
plt.xlabel('Label')
plt.title('Number of ham and spam messages')
```

/usr/local/lib/python3.7/dist-packages/seaborn/\_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.

FutureWarning

Out[35]:

Text(0.5, 1.0, 'Number of ham and spam messages')



In [36]:

```
X = df.v2
Y = df.v1
le = LabelEncoder()
Y = le.fit_transform(Y)
Y = Y.reshape(-1,1)
```

## Create model and 5.Add layers(LSTM,Dense-(Hidden layers),Output

In [11]:

In [37]:

```
inputs = Input(name='inputs', shape=[max_len])
layer = Embedding(max_words, 50, input_length=max_len)(inputs)
layer = LSTM(64)(layer)
layer = Dense(256, name='FC1')(layer)
layer = Activation('relu')(layer)
layer = Dropout(0.5)(layer)
layer = Dense(1, name='out_layer')(layer)
layer = Activation('sigmoid')(layer)
model = Model(inputs=inputs, outputs=layer)
model.summary()
```

Model: "model"

Layer (type)	Output Shape	Param #
=====		
inputs (InputLayer)	[(None, 150)]	0
embedding (Embedding)	(None, 150, 50)	50000
lstm (LSTM)	(None, 64)	29440
FC1 (Dense)	(None, 256)	16640
activation (Activation)	(None, 256)	0
dropout (Dropout)	(None, 256)	0
out_layer (Dense)	(None, 1)	257
activation_1 (Activation)	(None, 1)	0
=====		
Total params: 96,337		
Trainable params: 96,337		
Non-trainable params: 0		

## Compile the model

In [12]:

In [38]:

```
model.compile(optimizer='adam', loss='mse')
```

## Save the model

In [14]:

In [46]:

```
model.save('Spam_classifier.h5')
```