1. Importing the required library

```
import pandas as pd
import numpy as np
import nltk
import re  # Remove unwanted char.

nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from sklearn import preprocessing
from sklearn.feature_extraction.text import CountVectorizer

# ANN Model

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

2. Read dataset and do pre-processing

```
df = pd.read_csv('/content/spam.csv',encoding="ISO-8859-1")
df
```

	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

```
# df.loc[:, 'v2']
port = PorterStemmer()
data = []

for i in range(len(df)):
  review = df['v2'][i] # Reading data
  review = re.sub('[^a-zA-Z]',' ',review) # Removing special char.
  review = review.lower() # Convert capital letters into small letters
  review = review.split() # Split the input
  review = [port.stem(w) for w in review if w not in set(stopwords.words('english'))] # Stemming & Stopwords
  review = ' '.join(review) # join words
  data.append(review)

data
```

```
 white fudg oreo store ',
'januari male sale hot gay chat cheaper call nation rate p min cheap p min peak stop text call p min',
'love come took long leav zaher got word ym happi see sad left miss',
'sorri hurt',
'feel nauseou piss eat sweet week caus today plan pig diet week hungri',
'ok lor earli still project meet',
'call da wait call',
'could ask carlo could get anybodi els chip'
```

```
 'could ask carlo could get anybodi els chip',
 'actual send remind today wonder weekend',
 'peopl see msg think iam addict msging wrong bcoz know iam addict sweet friend bslvyl',
 'hey gave photo regist drive ah tmr wanna meet yck',
 'dont talk ever ok word',
 'u wana see',
 'way school pl send ashley number',
 'shall fine avalarr hollalat',
 'went attend anoth two round today still reach home',
 'actual delet old websit blog magicalsong blogspot com',
 'k wait chikku il send aftr lt gt min',
 'diet ate mani slice pizza yesterday ugh alway diet',
 'k give kvb acc detail',
 'oh come ah',
 'money r lucki winner claim prize text money million give away ppt x normal text rate box w jy',
 'realli sorri b abl friday hope u find altern hope yr term go ok',
 'congratul ore mo owo wa enjoy wish mani happi moment fro wherev go',
 'samu shoulder yet',
 'time think need know near campu',
 'dear matthew pleas call landlin complimentari lux tenerif holiday cash await collect ppm sae cs box sk xh',
 'dun wear jean lor',
 'sinc side fever vomitin',
 'k k colleg',
 'urgent call landlin complimentari tenerif holiday cash await collect sae cs box hp yf ppm',
 'better made friday stuf like pig yesterday feel bleh least writh pain kind bleh',
 'sell ton coin sell coin someon thru paypal voila money back life pocket',
 'theyr lot place hospit medic place safe',
 'get touch folk wait compani txt back name age opt enjoy commun p sm',
 'also sorta blown coupl time recent id rather text blue look weed',
 'sent score sopha secondari applic school think think appli research cost also contact joke ogunrind school one
less expens one',
 'cant wait see photo use',

 'ur cash balanc current pound maxim ur cash send go p msg cc po box tcr w',
 'hey book kb sat alreadi lesson go ah keep sat night free need meet confirm lodg',
 'chk ur belovd ms dict',
 'time want come',
 'awesom lemm know whenev around',
 'shb b ok lor thanx',
 'beauti truth graviti read care heart feel light someon feel heavi someon leav good night',
 'also rememb get dobbi bowl car',
 'filthi stori girl wait',
 'sorri c ur msg yar lor poor thing one night tmr u brand new room sleep',
 'love decis feel could decid love life would much simpler less magic'
```

```
        'welp appar retir',
        'sort code acc bank natwest repli confirm sent right person',
        '',
        'u sure u take sick time',
        'urgent tri contact u today draw show prize guarante call land line claim valid hr',
        'watch cartoon listen music amp eve go templ amp church u',
        'yo chad gymnast class wanna take site say christian class full',
        'much buzi',
```

```
cv = CountVectorizer()

x = cv.fit_transform(data).toarray()
x
len(x)
```

```
5572
```

```
# label_encoder object knows how to understand word labels.
label_encoder = preprocessing.LabelEncoder()

# Encode labels in column 'species'.
y = label_encoder.fit_transform(df['v1'])
y
```

```
array([0, 0, 1, ..., 0, 0, 0])
```

3. Create Model and Add Layers

```
# ANN Block

model = Sequential()
model.add(Dense(1500, activation='relu'))
model.add(Dense(3000, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
```

4. Compile the model

```
model.compile(optimizer='adam', loss='binary_crossentropy',metrics=['accuracy'])
```

5. Fit the model

```
model.fit(x,y,epochs=10)
```

```
Epoch 1/10
175/175 [==============================] - 18s 94ms/step - loss: 0.1137 - accuracy: 0.9659
Epoch 2/10
175/175 [==============================] - 16s 93ms/step - loss: 0.0105 - accuracy: 0.9973
Epoch 3/10
175/175 [==============================] - 15s 88ms/step - loss: 0.0016 - accuracy: 0.9991
Epoch 4/10
175/175 [==============================] - 15s 88ms/step - loss: 2.3136e-04 - accuracy: 1.0000
Epoch 5/10
175/175 [==============================] - 15s 88ms/step - loss: 8.2874e-05 - accuracy: 1.0000
Epoch 6/10
175/175 [==============================] - 15s 88ms/step - loss: 4.6353e-05 - accuracy: 1.0000
Epoch 7/10
175/175 [==============================] - 16s 89ms/step - loss: 2.8950e-05 - accuracy: 1.0000
Epoch 8/10
175/175 [==============================] - 15s 88ms/step - loss: 1.8780e-05 - accuracy: 1.0000
Epoch 9/10
175/175 [==============================] - 16s 93ms/step - loss: 1.2697e-05 - accuracy: 1.0000
Epoch 10/10
175/175 [==============================] - 15s 87ms/step - loss: 8.9532e-06 - accuracy: 1.0000
<keras.callbacks.History at 0x7f2211b63d90>
```

6. Save The Model

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

7. Test The Model

```
# Test 1

text =  'nah think goe usf live around though'
print(text)
print('*'*50)
text = re.sub('[^a-zA-Z]',' ',text)
print(text)
print('*'*50)
text = text.lower()
print(text)
print('*'*50)
text = text.split()
print(text)
print('*'*50)
text = [port.stem(w) for w in text if w not in set(stopwords.words('english'))]
print(text)
print('*'*50)
text = ' '.join(text)
print(text)
print('*'*50)
text = cv.transform([text]).toarray()
print(text)
print('*'*50)
pred = model.predict(text)
print(pred)
print('*'*50)
if pred>0.5:
  print('Positive')
else: print('Negative')
```

```
nah think goe usf live around though
**************************************************
nah think goe usf live around though
**************************************************
nah think goe usf live around though
**************************************************
['nah', 'think', 'goe', 'usf', 'live', 'around', 'though']
**************************************************
['nah', 'think', 'goe', 'usf', 'live', 'around', 'though']
**************************************************
nah think goe usf live around though
**************************************************
[[0 0 0 ... 0 0 0]]
**************************************************
1/1 [==============================] - 0s 73ms/step
[[3.562159e-11]]
**************************************************
Negative
```

```
# Test 2

text = 'hi im relax time ever get everi day parti good night get home tomorrow ish'
print(text)
print('*'*50)
text = re.sub('[^a-zA-Z]',' ',text)
print(text)
print('*'*50)
text = text.lower()
print(text)
print('*'*50)
text = text.split()
print(text)
print('*'*50)
text = [port.stem(w) for w in text if w not in set(stopwords.words('english'))]
print(text)
print('*'*50)
text = ' '.join(text)
print(text)
print('*'*50)
text = cv.transform([text]).toarray()
```

```
print(text)
print('*'*50)
pred = model.predict(text)
print(pred)
print('*'*50)
if pred>0.5:
  print('Positive')
else: print('Negative')
```

```
hi im relax time ever get everi day parti good night get home tomorrow ish
**************************************************
hi im relax time ever get everi day parti good night get home tomorrow ish
**************************************************
hi im relax time ever get everi day parti good night get home tomorrow ish
**************************************************
['hi', 'im', 'relax', 'time', 'ever', 'get', 'everi', 'day', 'parti', 'good', 'night', 'get', 'home', 'tomorrow', 'ish'
**************************************************
['hi', 'im', 'relax', 'time', 'ever', 'get', 'everi', 'day', 'parti', 'good', 'night', 'get', 'home', 'tomorrow', 'ish'
**************************************************
hi im relax time ever get everi day parti good night get home tomorrow ish
**************************************************
[[0 0 0 ... 0 0 0]]
**************************************************
1/1 [==============================] - 0s 18ms/step
[[2.4121484e-19]]
**************************************************
Negative
```

Colab paid products - Cancel contracts here

✓ 0s completed at 8:39 AM