```
import numpy as np
import pandas as pd
import re
import nltk
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from sklearn.feature_extraction.text import CountVectorizer
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
from sklearn.model_selection import train_test_split


df=pd.read_csv(r"spam.csv",encoding='Windows-1252')
df.head()
```

	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

```
df.describe()
```

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
count	5572	5572	50	12	6
unique	2	5169	43	10	5
top	ham	Sorry, I'll call later	bt not his girlfrnd... G o o d n i g h t . . .@"	MK17 92H. 450Ppw 16"	GNT:-)"

```
ps=PorterStemmer()
nltk.download('stopwords')
```

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

```
data=[]
for i in range(0,5572):
    message=df["v2"][i]
    message=message.lower()
    message=re.sub('[^a-z]',' ',message)
    message=message.split()
    message=[ps.stem(word) for word in message if not word in set(stopwords.words("english
```

```
    message=' '.join(message)
    data.append(message)

data
```

```
['go jurong point crazi avail bugi n great world la e buffet cine got amor wat',
 'ok lar joke wif u oni',
 'free entri wkli comp win fa cup final tkt st may text fa receiv entri question
std txt rate c appli',
 'u dun say earli hor u c alreadi say',
 'nah think goe usf live around though',
 'freemsg hey darl week word back like fun still tb ok xxx std chg send rcv',
 'even brother like speak treat like aid patent',
 'per request mell mell oru minnaminungint nurungu vettam set callertun caller
press copi friend callertun',
 'winner valu network custom select receivea prize reward claim call claim code kl
valid hour',
 'mobil month u r entitl updat latest colour mobil camera free call mobil updat co
free',
 'gonna home soon want talk stuff anymor tonight k cri enough today',
 'six chanc win cash pound txt csh send cost p day day tsandc appli repli hl
info',
 'urgent week free membership prize jackpot txt word claim c www dbuk net lccltd
pobox ldnw rw',
 'search right word thank breather promis wont take help grant fulfil promis
wonder bless time',
 'date sunday',
 'xxxmobilemovieclub use credit click wap link next txt messag click http wap
xxxmobilemovieclub com n qjkgighjjgcbl',
 'oh k watch',
 'eh u rememb spell name ye v naughti make v wet',
 'fine way u feel way gota b',
 'england v macedonia dont miss goal team news txt ur nation team eg england tri
wale scotland txt poboxox w wq',
 'serious spell name',
 'go tri month ha ha joke',
 'pay first lar da stock comin',
 'aft finish lunch go str lor ard smth lor u finish ur lunch alreadi',
 'fffffffff alright way meet sooner',
 'forc eat slice realli hungri tho suck mark get worri know sick turn pizza lol',
 'lol alway convinc',
 'catch bu fri egg make tea eat mom left dinner feel love',
 'back amp pack car let know room',
 'ahhh work vagu rememb feel like lol',
 'wait still clear sure sarcast x want live us',
 'yeah got v apologet n fallen actin like spoilt child got caught till go badli
cheer',
 'k tell anyth',
 'fear faint housework quick cuppa',
 'thank subscript rington uk mobil charg month pleas confirm repli ye repli
charg',
 'yup ok go home look time msg xuhui go learn nd may lesson',
 'oop let know roommat done',
 'see letter b car',
 'anyth lor u decid',
 'hello saturday go text see decid anyth tomo tri invit anyth',
 'pl go ahead watt want sure great weekend abiola',
 'forget tell want need crave love sweet arabian steed mmmmmm yummi',
 'rodger burn msg tri call repli sm free nokia mobil free camcord pleas call
deliveri tomorrow',
```

```
        'see',
        'great hope like man well endow lt gt inch',
        'call messag miss call',
```

```
cv=CountVectorizer(max_features=7000)
x=cv.fit_transform(data).toarray()
x.shape
```

```
(5572, 6221)
```

```
df["v1"].loc[df["v1"]=="spam"]=0.0
df["v1"].loc[df["v1"]=="ham"]=1.0
df["v1"]
```

```
0       1.0
1       1.0
2       0.0
3       1.0
4       1.0
       ...
5567    0.0
5568    1.0
5569    1.0
5570    1.0
5571    1.0
Name: v1, Length: 5572, dtype: object
```

```
y=df.iloc[:,0:1].values
y=np.asarray(y).astype("float64")
y
```

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

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
```

```
model=Sequential()
```

```
model.add(Dense(units=5572,activation='relu',kernel_initializer='random_uniform'))
#hidden layer
model.add(Dense(units=6000,activation='relu',kernel_initializer='random_uniform'))
model.add(Dense(units=6000,activation='relu',kernel_initializer='random_uniform'))
model.add(Dense(units=6000,activation='relu',kernel_initializer='random_uniform'))
model.add(Dense(units=6000,activation='relu',kernel_initializer='random_uniform'))
model.add(Dense(units=6000,activation='relu',kernel_initializer='random_uniform'))
#output layer
model.add(Dense(units=1,activation='sigmoid',kernel_initializer='random_uniform'))
```

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


tr=model.fit(x_train,y_train,epochs=10,batch_size=32)
```

```
Epoch 1/10
140/140 [==============================] - 252s 2s/step - loss: 1.7631 - accuracy: 0
Epoch 2/10
140/140 [==============================] - 241s 2s/step - loss: 0.0611 - accuracy: 0
Epoch 3/10
140/140 [==============================] - 247s 2s/step - loss: 0.6032 - accuracy: 0
Epoch 4/10
140/140 [==============================] - 247s 2s/step - loss: 0.4485 - accuracy: 0
Epoch 5/10
140/140 [==============================] - 247s 2s/step - loss: 0.1708 - accuracy: 0
Epoch 6/10
140/140 [==============================] - 246s 2s/step - loss: 2.3482 - accuracy: 0
Epoch 7/10
140/140 [==============================] - 246s 2s/step - loss: 0.1099 - accuracy: 0
Epoch 8/10
140/140 [==============================] - 246s 2s/step - loss: 0.0037 - accuracy: 0
Epoch 9/10
140/140 [==============================] - 246s 2s/step - loss: 0.0033 - accuracy: 0
Epoch 10/10
140/140 [==============================] - 247s 2s/step - loss: 0.1228 - accuracy: 0
```

```
model.save("sms.h5")


ypred=model.predict(x_test)
ypred
```

```
35/35 [==============================] - 15s 399ms/step
array([[1.],
       [1.],
       [1.],
       ...,
       [1.],
       [1.],
       [1.]], dtype=float32)
```

```
y_test
```

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

```
text=model.predict(cv.transform(["Wishing you a very happy Birthday to you ! "]))
text>0.5
```

```
1/1 [==============================] - 0s 305ms/step
array([[ True]])
```

```
class_name=["ham","spam"]
pred_id=text.argmax(axis=1)[0]
pred_id
print(str(class_name[pred_id]))
```

```
ham
```