Assignment-4

|  |  |
| --- | --- |
| Assignment Date | 16 november 2022 |
| Student Name | S.kiruthika |
| Student Roll Number | 421319104014 |
| Maximum Marks | 2Marks |

Question 1:

Download the dataset Link:

[https://drive.google.com/file/d/1Sjqx5H5R86tRp2YZKzzd4\_iEfjChZ3ob/](https://drive.google.com/file/d/1Sjqx5H5R86tRp2YZKzzd4_iEfjChZ3ob/view?usp=sharing) [view?usp=sharing](https://drive.google.com/file/d/1Sjqx5H5R86tRp2YZKzzd4_iEfjChZ3ob/view?usp=sharing)

Question 2:

Import required library Solution:

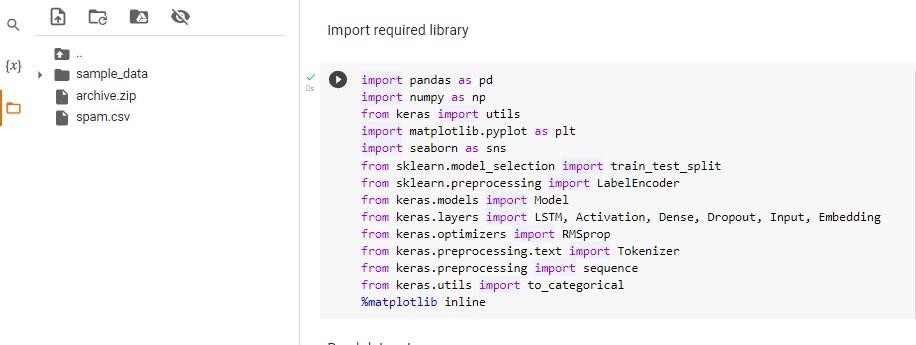
import pandas as pd import numpy as np from keras import utils

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



Question 3:

Read dataset and do pre-processing Solution:

Read dataset

# !unzip "/content/archive.zip"

df = pd.read\_csv('spam.csv',delimiter=',',encoding='latin-1') df

Pre processing

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

sns.countplot(df.v1,palette='Set3') plt.xlabel('Label')

plt.title('Number of ham and spam messages')

X = df.v2 Y = df.v1

le = LabelEncoder() Y = le.fit\_transform(Y) Y = Y.reshape(-1,1)

X\_train,X\_test,Y\_train,Y\_test = train\_test\_split(X,Y,test\_size=0.15) max\_words = 1000

max\_len = 150

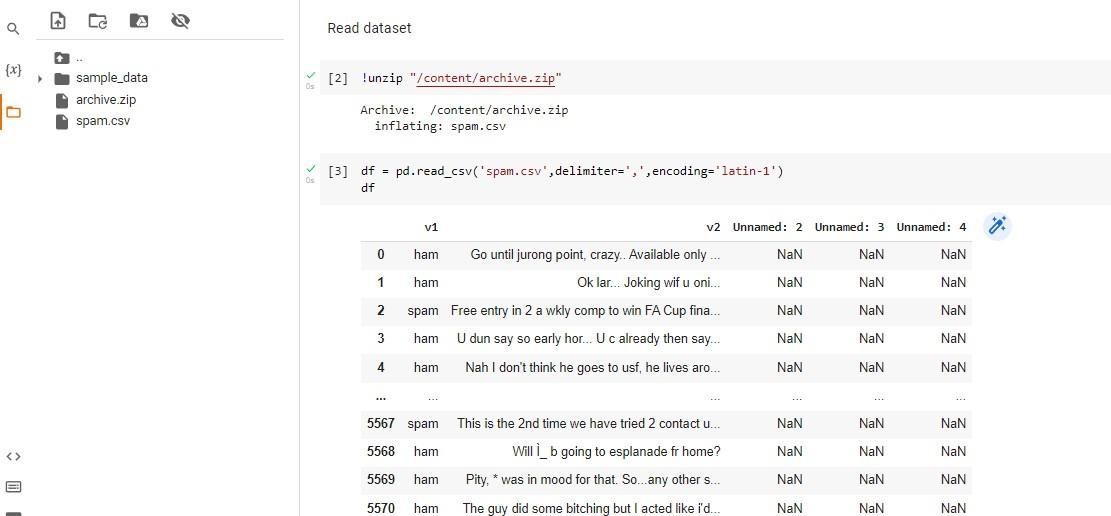
tok = Tokenizer(num\_words=max\_words) tok.fit\_on\_texts(X\_train)

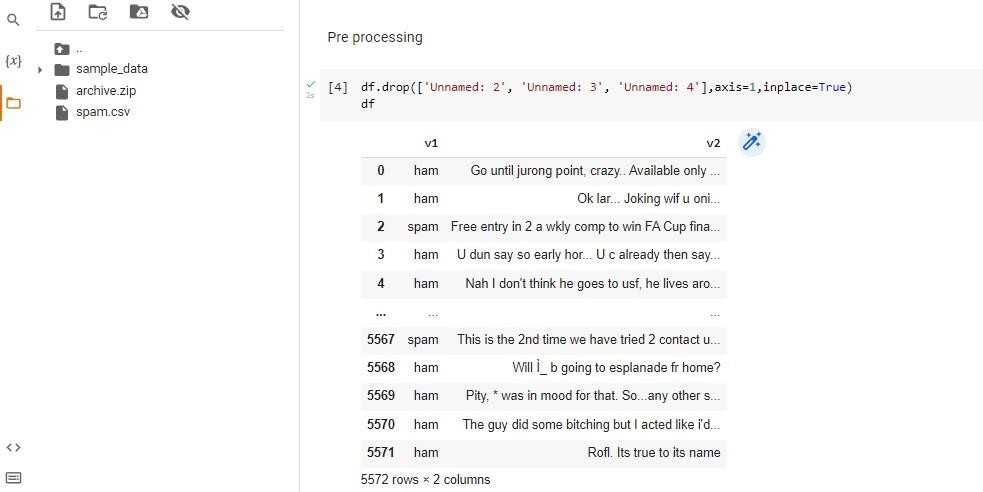
sequences = tok.texts\_to\_sequences(X\_train)

sequences\_matrix = utils.pad\_sequences(sequences,maxlen=max\_len)

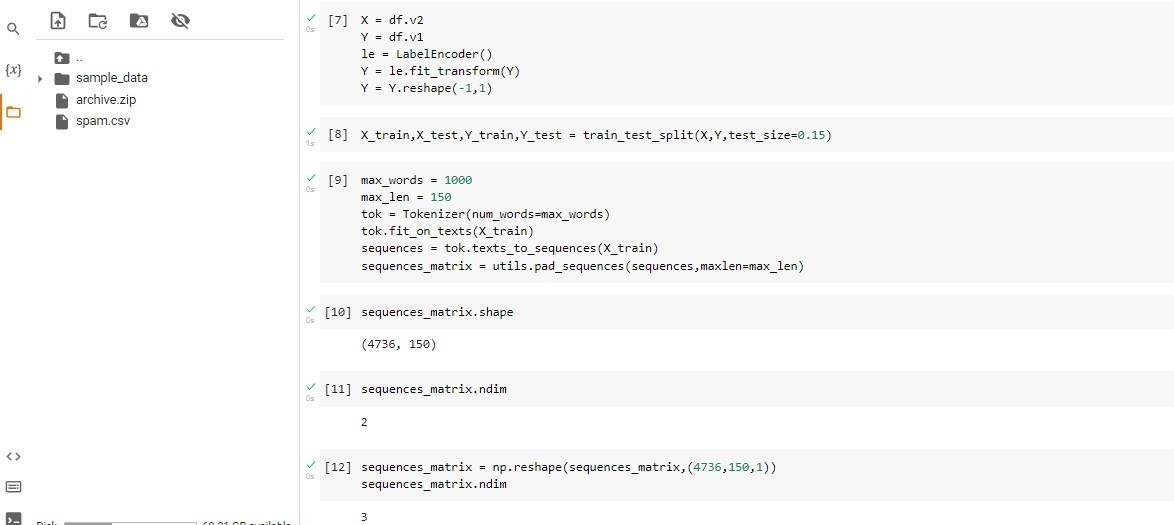
sequences\_matrix.shape sequences\_matrix.ndim

sequences\_matrix = np.reshape(sequences\_matrix,(4736,150,1)) sequences\_matrix.ndim









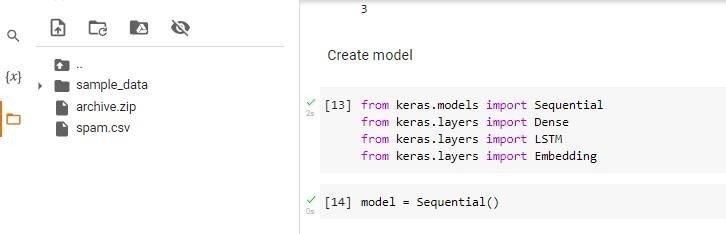
Question 4:

Create model Solution:

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

from keras.layers import LSTM

from keras.layers import Embedding model = Sequential()

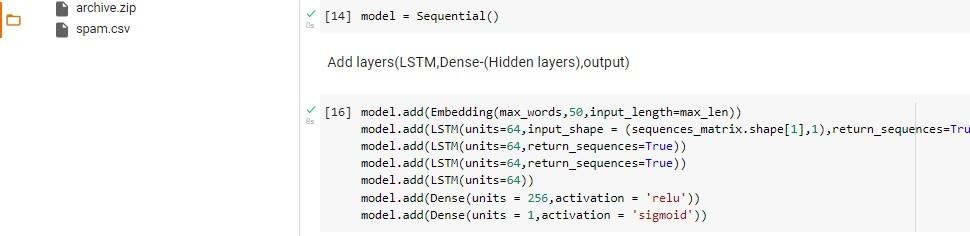


Question 5:

Add layers(LSTM,Dense-(Hidden layers),output) Solution:

model.add(Embedding(max\_words,50,input\_length=max\_len)) model.add(LSTM(units=64,input\_shape = (sequences\_matrix.shape[1],1),return\_sequences=True)) model.add(LSTM(units=64,return\_sequences=True)) model.add(LSTM(units=64,return\_sequences=True))

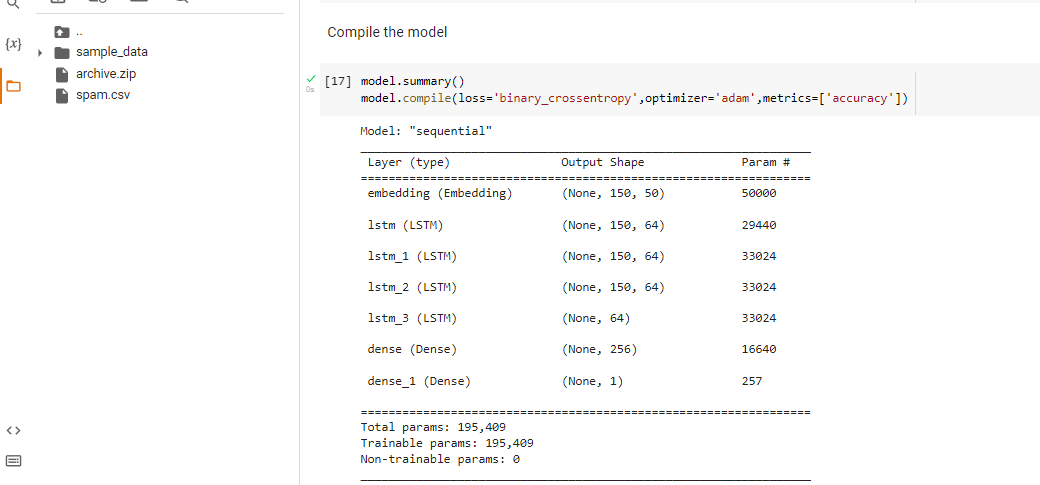
model.add(LSTM(units=64)) model.add(Dense(units = 256,activation = 'relu')) model.add(Dense(units = 1,activation = 'sigmoid'))



Question 6:

Compile the model Solution:

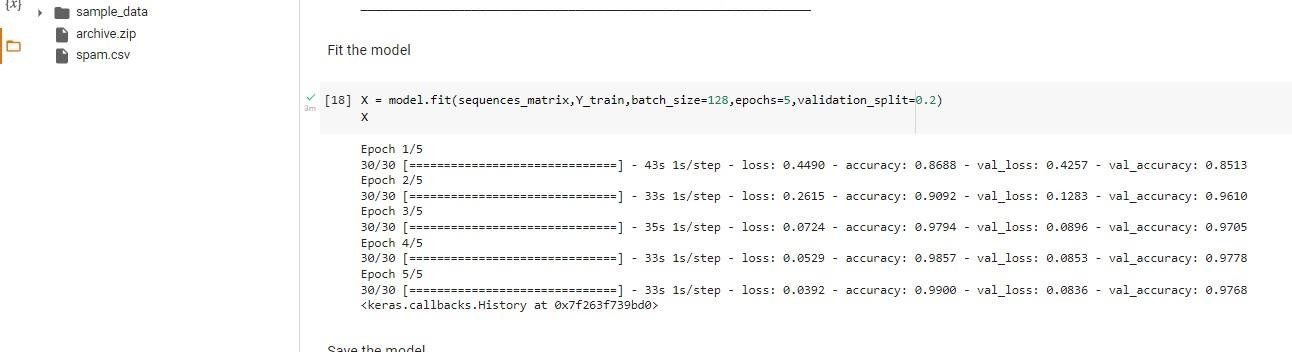
model.summary() model.compile(loss='binary\_crossentropy',optimizer='adam',metrics=['accuracy'])



Question 7:

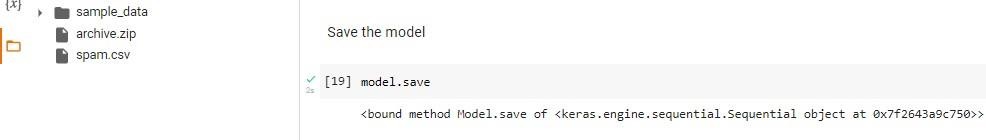
Fit the model Solution:

X = model.fit(sequences\_matrix,Y\_train,batch\_size=128,epochs=5,validation\_split=0.2) X



Question 8: Save the model Solution:

model.save



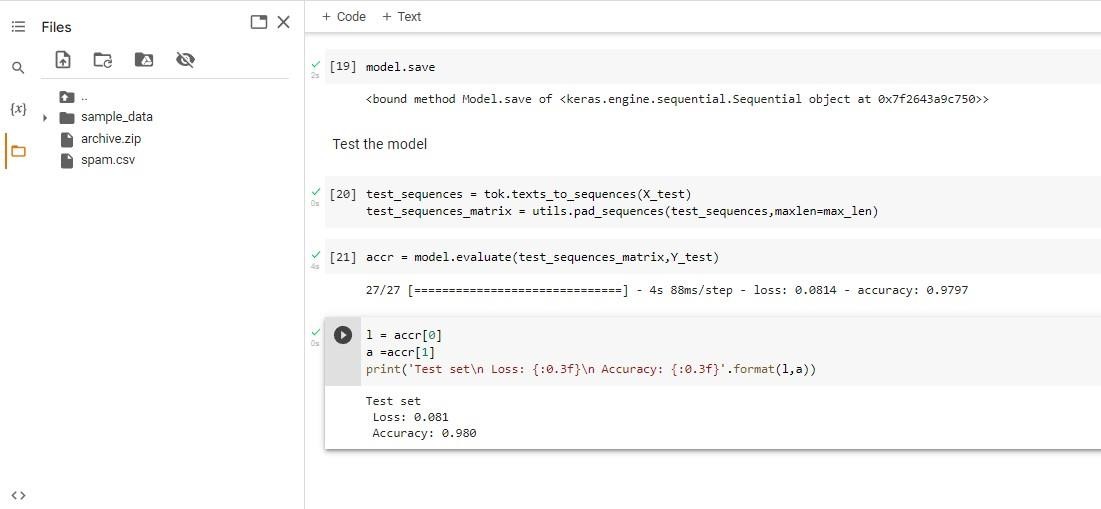
Question 9:

Test the model Solution:

test\_sequences = tok.texts\_to\_sequences(X\_test)

test\_sequences\_matrix = utils.pad\_sequences(test\_sequences,maxlen=max\_len) accr = model.evaluate(test\_sequences\_matrix,Y\_test)

l = accr[0] a =accr[1]

print('Test set\n Loss: {:0.3f}\n Accuracy: {:0.3f}'.format(l,a))