Assignment-4

|  |  |
| --- | --- |
| AssignmentDate | 17 november2022 |
| StudentName | M.Eashwar |
| StudentRollNumber | 421319104004 |
| MaximumMarks | 2Marks |

Question1:

DownloadthedatasetLink:

<https://drive.google.com/file/d/1Sjqx5H5R86tRp2YZKzzd4_iEfjChZ3ob/view?usp=sharing>

Question2:

ImportrequiredlibrarySolution:

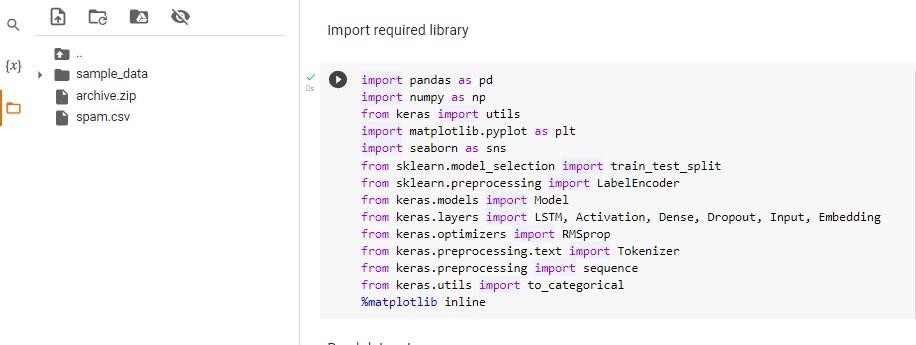
importpandas as pdimport numpyas npfromkerasimportutils

importmatplotlib.pyplotaspltimportseabornassns

fromsklearn.model\_selectionimporttrain\_test\_splitfrom sklearn.preprocessingimport LabelEncoderfromkeras.modelsimportModel

fromkeras.layersimportLSTM,Activation,Dense,Dropout,Input,Embeddingfromkeras.optimizersimportRMSprop

fromkeras.preprocessing.textimportTokenizerfrom keras.preprocessingimport sequencefromkeras.utilsimportto\_categorical



Question3:

Readdatasetanddopre-processingSolution:

Readdataset

# !unzip"/content/archive.zip"

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

Preprocessing

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

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

plt.title('Numberofhamandspammessages')

X = df.v2Y=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

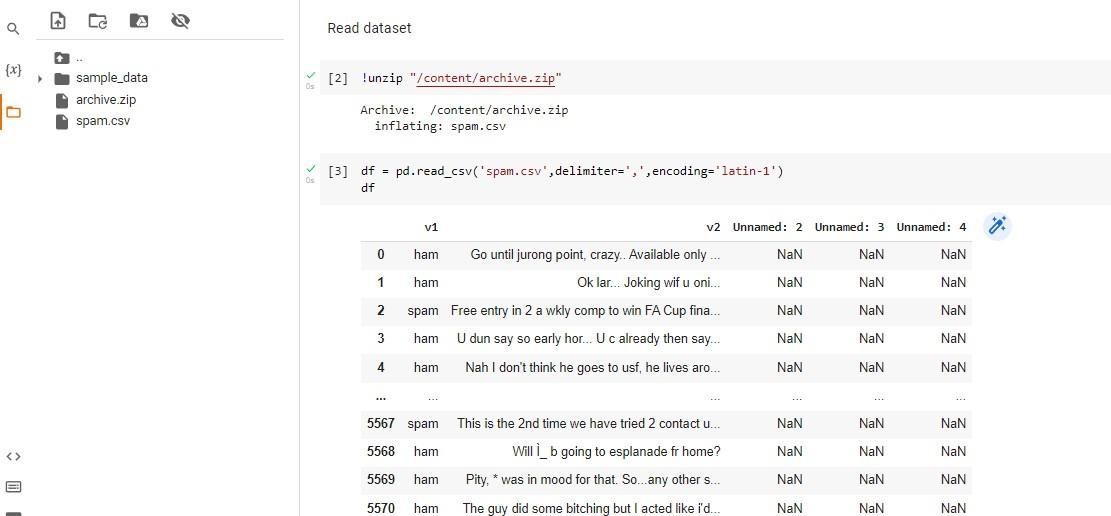
tok= Tokenizer(num\_words=max\_words)tok.fit\_on\_texts(X\_train)

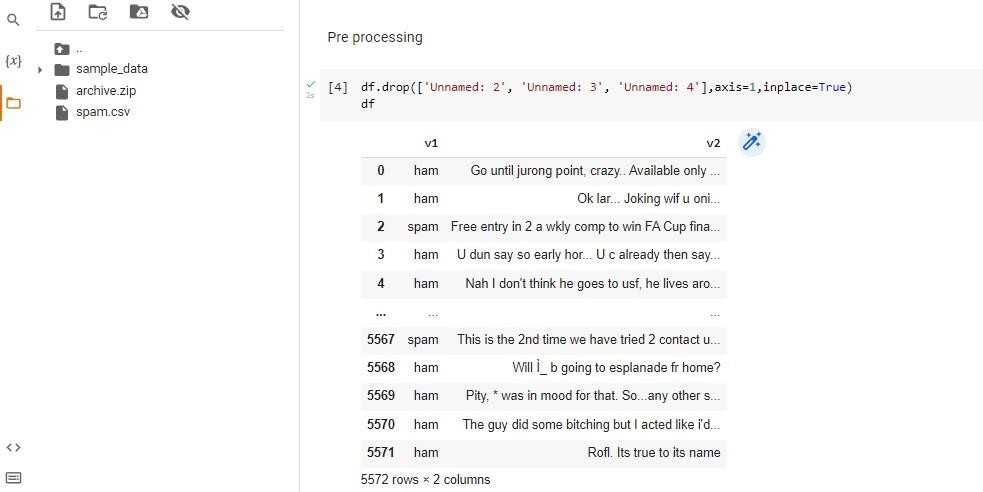
sequences=tok.texts\_to\_sequences(X\_train)

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

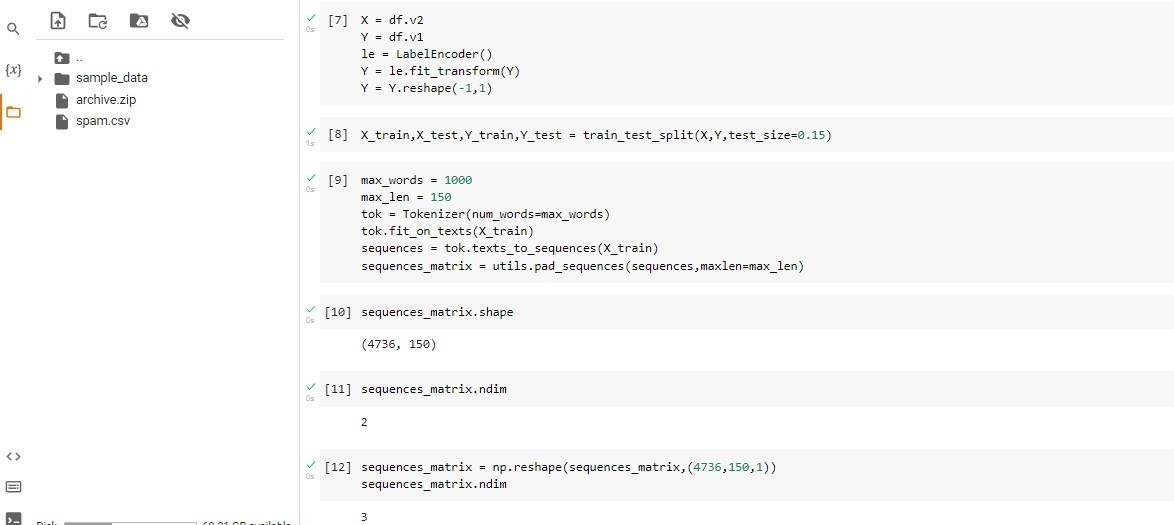
sequences\_matrix.shapesequences\_matrix.ndim

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









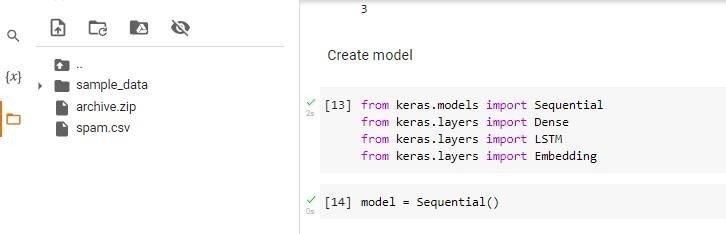
Question4:

Create modelSolution:

fromkeras.modelsimportSequentialfromkeras.layersimportDense

fromkeras.layersimportLSTM

fromkeras.layersimportEmbeddingmodel=Sequential()

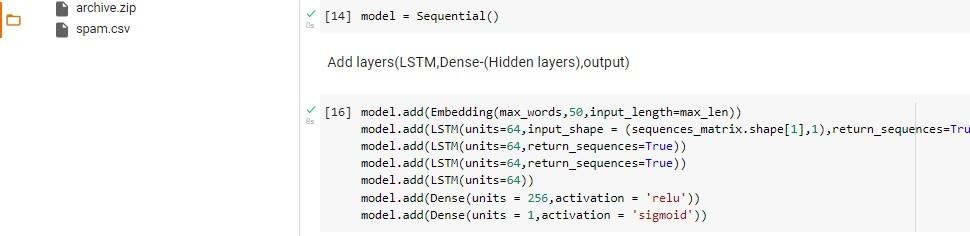


Question5:

Addlayers(LSTM,Dense-(Hiddenlayers),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))

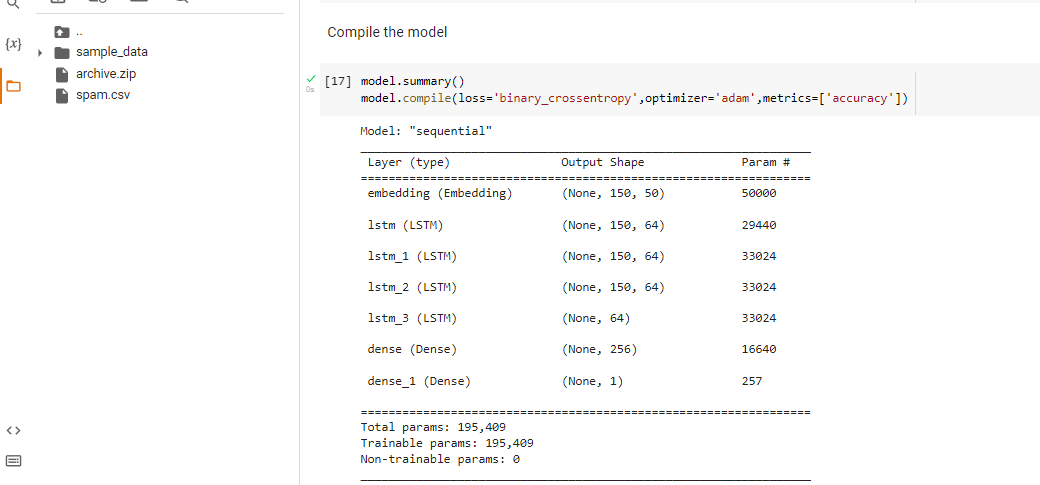
model.add(LSTM(units=64))model.add(Dense(units = 256,activation = 'relu'))model.add(Dense(units=1,activation='sigmoid'))



Question6:

CompilethemodelSolution:

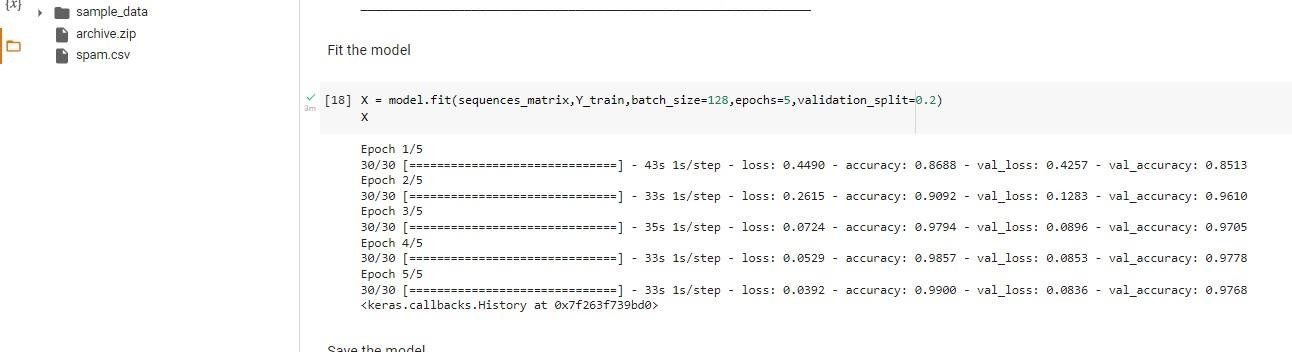
model.summary()model.compile(loss='binary\_crossentropy',optimizer='adam',metrics=['accuracy'])



Question7:

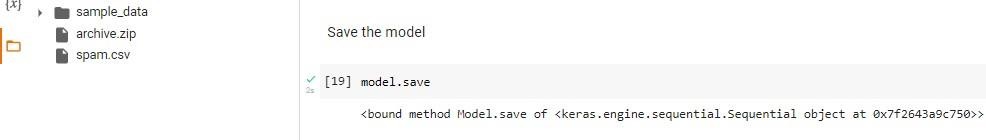
FitthemodelSolution:

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



Question8:SavethemodelSolution:

model.save



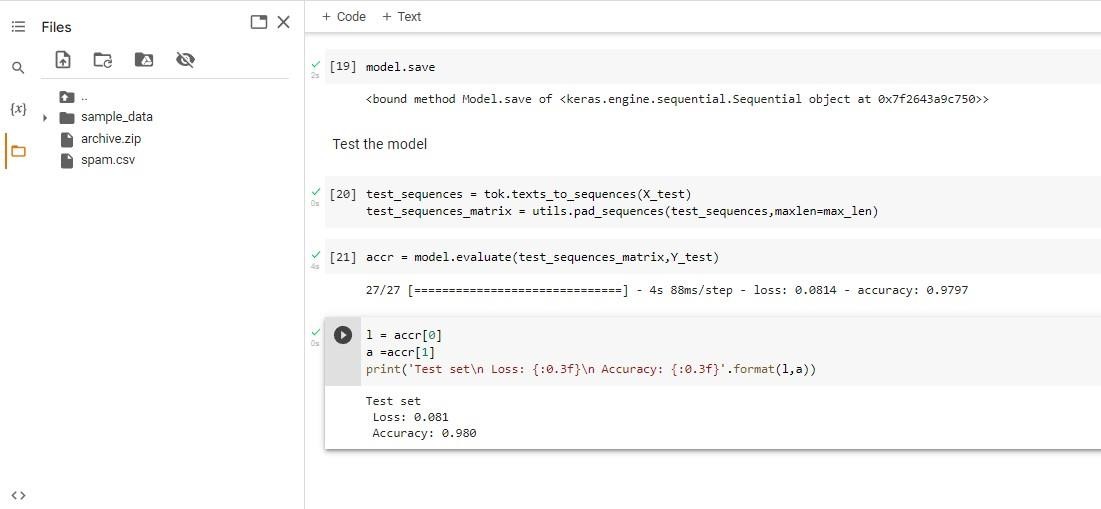
Question9:

TestthemodelSolution:

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('Testset\nLoss:{:0.3f}\nAccuracy: {:0.3f}'.format(l,a))