FASHION

import numpy as np

from keras.utils import np\_utils

from keras.datasets import fashion\_mnist

from keras.models import Sequential

from keras.layers import Dense, Dropout

from tensorflow.keras.optimizers import RMSprop

import matplotlib.pyplot as plt

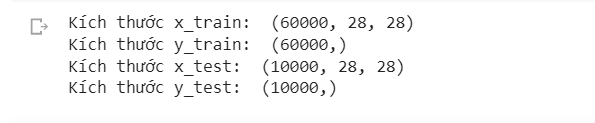
(x\_train, y\_train),(x\_test,y\_test) = fashion\_mnist.load\_data()

print('Kích thước x\_train: ', x\_train.shape)

print('Kích thước y\_train: ', y\_train.shape)

print('Kích thước x\_test: ', x\_test.shape)

print('Kích thước y\_test: ', y\_test.shape)



x\_train = x\_train.reshape(60000,784)

x\_test = x\_test.reshape(10000,784)

x\_train = x\_train.astype('float32')

x\_test = x\_test.astype('float32')

x\_train /=255

x\_test /=255

y\_train = np\_utils.to\_categorical(y\_train,10)

y\_test = np\_utils.to\_categorical(y\_test,10)

model = Sequential()

model.add(Dense(512,activation='relu',input\_shape=(784,)))

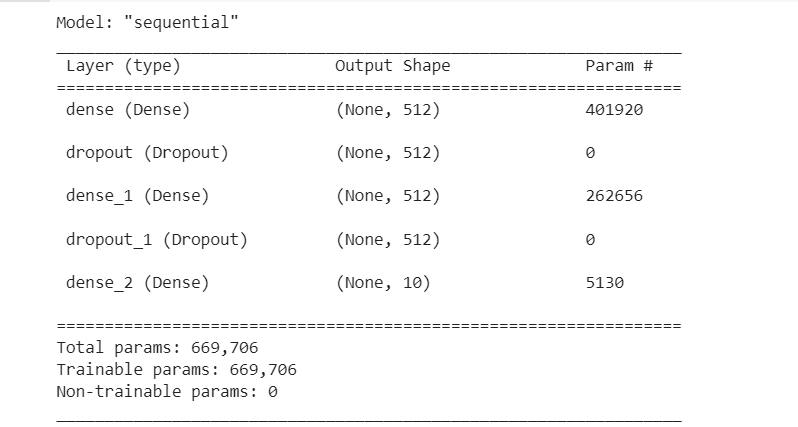
model.add(Dropout(0.2))

model.add(Dense(512,activation='relu'))

model.add(Dropout(0.2))

model.add(Dense(10,activation='softmax'))

model.summary()



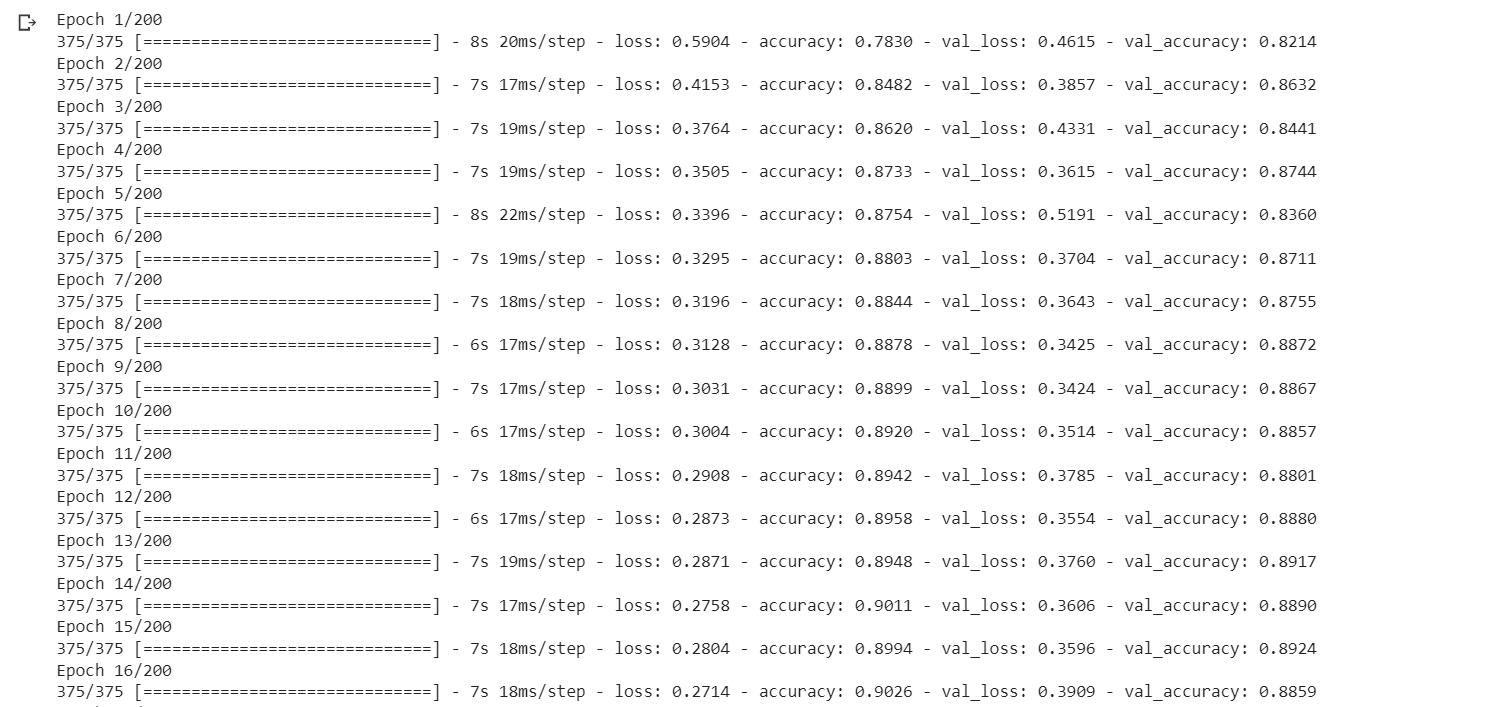
model.compile(loss='categorical\_crossentropy',optimizer=RMSprop(), metrics=['accuracy'])

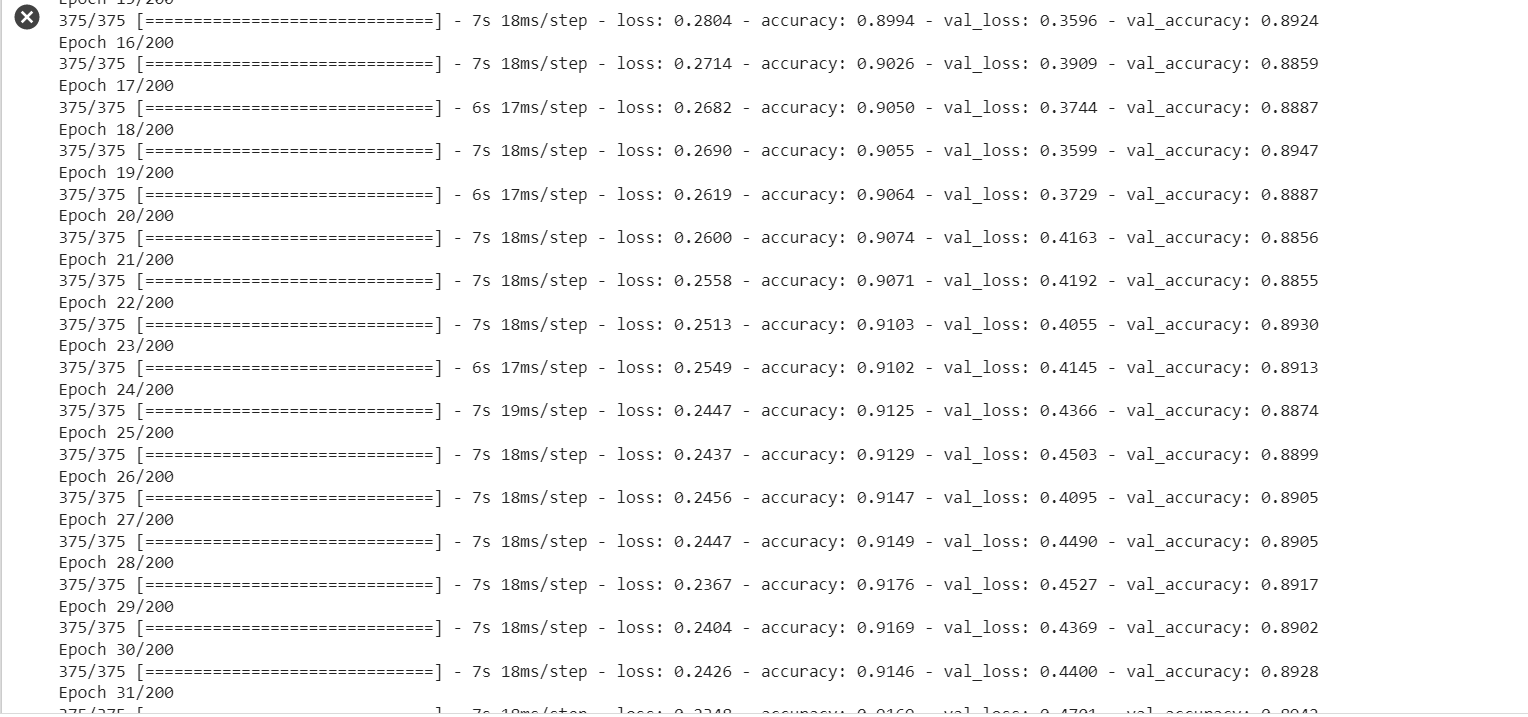
model.compile(loss='categorical\_crossentropy',optimizer=RMSprop(), metrics=['accuracy'])

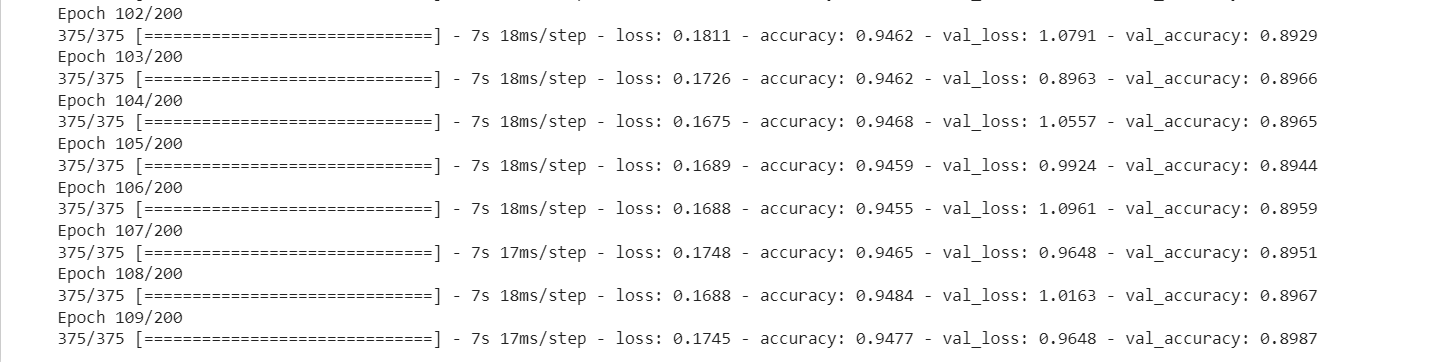
from keras.callbacks import EarlyStopping

history = model.fit(x\_train,y\_train,batch\_size=128,epochs=200,verbose=1,

                    validation\_split=0.2,callbacks=[EarlyStopping(monitor='val\_loss',patience=100)])







score = model.evaluate(x\_test,y\_test,verbose=0)

print('Sai số kiểm tra là: ',score[0])

print('Độ chính xác kiểm tra là: ',score[1])

#vẽ đồ thị

plt.plot(history.history['accuracy'])

plt.plot(history.history['val\_accuracy'])

plt.title('model accuracy')

plt.ylabel('accuracy')

plt.xlabel('epochs')

plt.legend(['train','Validation'])

plt.show()

