Exercise RNN

Create a language model using RNN. You can use the modelbuild method as below.

For the dataset you can choose any text corpus. Given a sequence of words, the model should predict the next word.

def modelbuild():

model = Sequential()

model.add(keras.layers.InputLayer(input\_shape=(15,1)))

keras.layers.embeddings.Embedding(nb\_words, 15, weights=[embedding\_matrix], input\_length=15,

trainable=False)

model.add(keras.layers.recurrent.SimpleRNN(units = 100, activation='relu',

use\_bias=True))

    model.add(keras.layers.Dense(units=1000, input\_dim = 2000, activation='sigmoid'))

    model.add(keras.layers.Dense(units=500, input\_dim=1000, activation='relu'))

    model.add(keras.layers.Dense(units=2, input\_dim=500,activation='softmax'))

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

return model

#compiling the model

finalmodel = modelbuild()

finalmodel.fit(trainx, trainy, epochs=10, batch\_size=120,validation\_data=(validx,validy))