

Lab 12 RNN

import numpy as np

import tensorflow as tf

idx2char = ['h', 'i', 'e', 'l', 'o']

Teach hello: hihell -> ihello

x_data = [[0, 1, 0, 2, 3, 3]] # hihell

y_data = [[1, 0, 2, 3, 3, 4]] # ihello

num_classes = 5

input_dim = 5 # one-hot size, same as hidden_size to directly predict one-hot

sequence_length = 6 # |ihello| == 6

learning_rate = 0.1 ↳ 문자의 개수

x_one_hot = np.array([[[[1, 0, 0, 0, 0], # h 0

[0, 1, 0, 0, 0], # i 1

[1, 0, 0, 0, 0], # h 0

[0, 0, 1, 0, 0], # e 2

[0, 0, 0, 1, 0], # l 3

[0, 0, 0, 1, 0]], # l 3

dtype=np.float32)

↳ one-hot 인코딩을 해주는 함수

y_one_hot = tf.keras.utils.to_categorical(y_data, num_classes=num_classes)

print(x_one_hot.shape)

print(y_one_hot.shape)

ex) shape(3, 5, 2)

[[[x,x],[x,x],[x,x],[x,x],[x,x]],

[[x,x],[x,x],[x,x],[x,x],[x,x]],

[[x,x],[x,x],[x,x],[x,x],[x,x]]]

tf.model = tf.keras.Sequential()

↳ batch size

make cell and add it to RNN layer

input_shape = (1,6,5) => number of sequence (batch), length of sequence, size of input dim

cell = tf.keras.layers.LSTMCell(units=num_classes, input_shape=(sequence_length, input_dim)) ↳ LSTM Cell을 생성

tf.model.add(tf.keras.layers.RNN(cell=cell, return_sequences=True))

RNN 셀 인스턴스 시퀀스의 마지막 출력을 반환할지, 아니면 전체 시퀀스를 반환할지

single LSTM layer can be used as well instead of creating LSTMCell

tf.model.add(tf.keras.layers.LSTM(units=num_classes, input_shape=(sequence_length, input_dim), return_sequences=True))

fully connected layer

tf.model.add(tf.keras.layers.TimeDistributed(tf.keras.layers.Dense(units=num_classes, activation='softmax')))

input의 모든 시간적 조각에 대해서 레이어를 적용합니다

tf.model.compile(loss='categorical_crossentropy', optimizer=tf.keras.optimizers.Adam(lr=learning_rate), metrics=['accuracy'])

train

tf.model.fit(x_one_hot, y_one_hot, epochs=50)

tf.model.summary()

predictions = tf.model.predict(x_one_hot)

for i, prediction in enumerate(predictions):

print(prediction) ↳ 순서와 리스트 값을 전달하는 기능을 가짐

print char using argmax, dict

```
result_str = [idx2char[c] for c in np.argmax(prediction, axis=1)]
print("\tPrediction str: ", ''.join(result_str))
```

↓
인덱스를 문자로 바꿔줌
↳ Prediction str: ihello

↳ axis=1 인 값들 중
가장 큰 값의 인덱스들을
반환하는 함수