조성현 강사님 강의.

LSTM 텍스트 자동 생성

> 자연어 처리 쿡북 *pp.309~316*

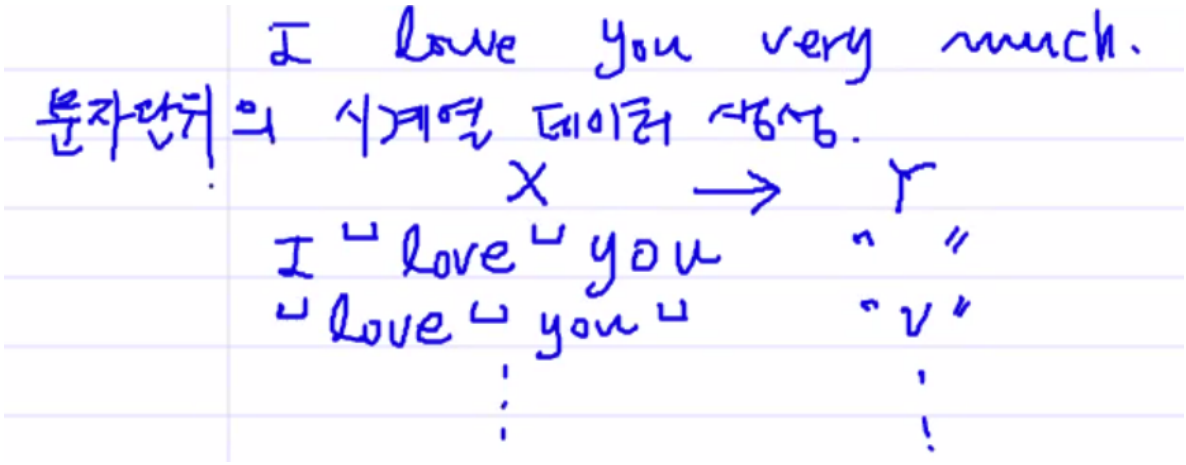
주어진 길이의 문장을 문자 단위로 학습하여 다음 문자로 예측한다. 텍스트를 자동화하여 연속적으로 생성한다.

1. 개요

LSTM 모델을 이용해 셰익스피어의 소설을 읽고, 학습한다. 공백 문자까지 하나의 문자로 포함한다.

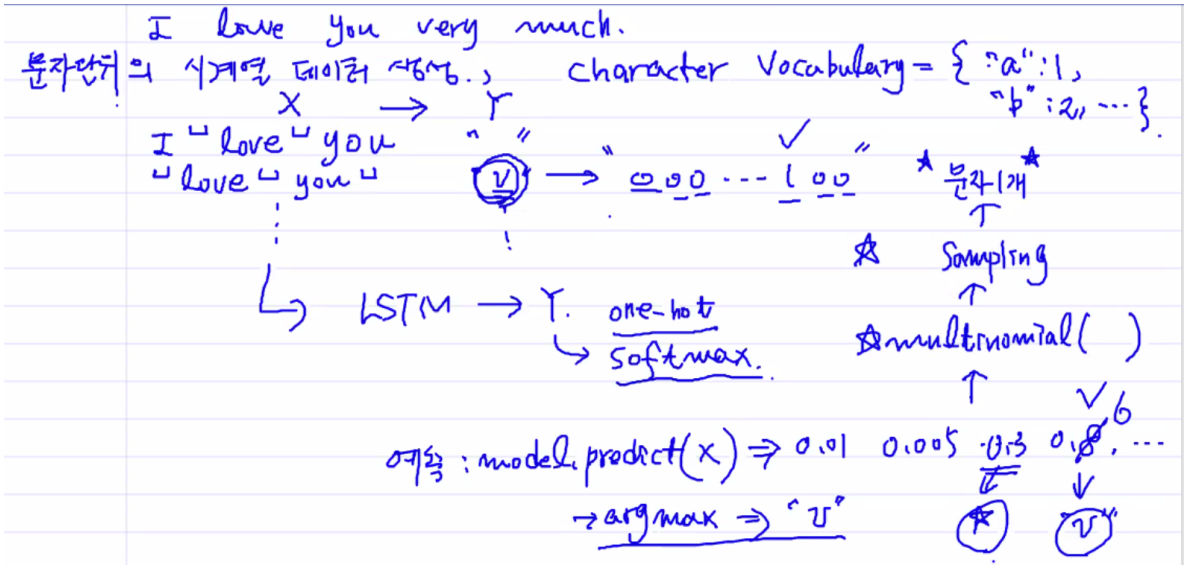
- `x` : 일정 길이의 문장
- `y` : 다음에 나올 문자

`x`에 들어갈 문자의 길이를 하나씩 뒤로 *shift* 하면서 **문자 단위의 시계열 데이터**를 만든다. `x`의 문장이 입력되면 `y`의 문자가 나오는 방식이다.



이 때 어휘 집합(Vocabulary)은 단어가 아닌, 문자 단위이다. character 단위로 `{'a' : 1, 'b' : 2, ...}` 와 같이 사전을 만들고, 이를 바탕으로 원핫 인코딩한다. 각 문장의 **문자**들이 모두 *원핫 벡터*가 되고, 다음에 나올 output **문자** 역시 *원핫 벡터*가 된다.

LSTM 모델을 구성하여 학습한다. 그리고 예측한다. `model.predict`를 통해 `softmax`가 취해진 예측값이 나온다.

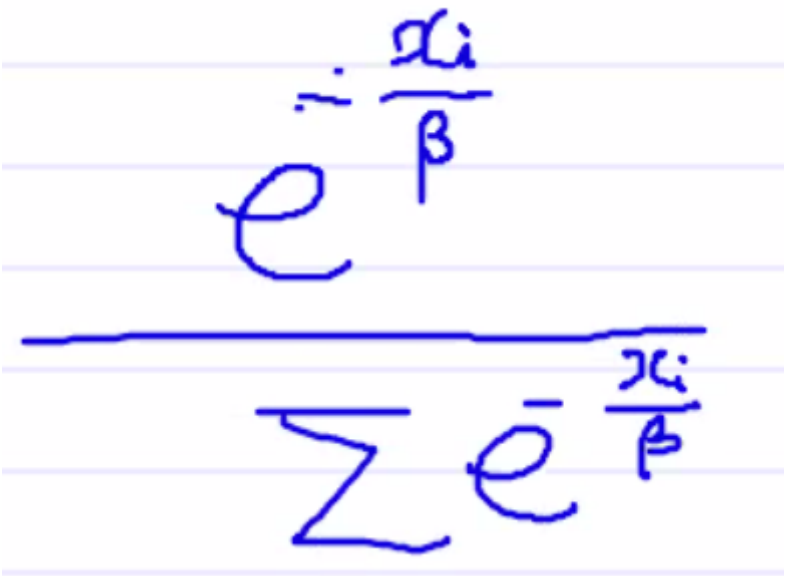


생성될 문자에 다양성 부여

기존에는 이렇게 나온 예측값에 `argmax`를 통해 인덱스 값을 뽑아 낸다. 그런데 예측되어 나올 문자에 다양성을 주기 위해 다음과 같은 방식을 사용한다.

Softmax β 조절

기존의 standard softmax 함수는 사실, 아래와 같은 softmax 함수에서 β가 1일 때의 함수이다.



β 의 크기를 조절함으로써 softmax 확률값의 차이가 달라지게 할 수 있다. 작을수록 softmax 확률값의 차이가 커진다.

아래의 간단한 예시를 통해 확인할 수 있다.

```
import numpy as np

a = np.array([0.6, 0.4])
beta = [0.2, 1.0, 5.0]
for b in beta:
    e = np.exp(a/b)
    softmax = e / np.sum(e)
    print(f"exponential : {e}")
    print(f"softmax : {softmax}")
```

```
exponential : [20.08553692  7.3890561 ]
softmax : [0.73105858 0.26894142]
exponential : [1.8221188 1.4918247]
softmax : [0.549834 0.450166]
exponential : [1.12749685 1.08328707]
softmax : [0.50999867 0.49000133]
```

β 를 조절하는 정도에 따라 softmax 로짓 값이 다르게 도출된다. 각 값의 차이가 크면 클수록 다양하게 선택될 확률이 높다.

> 챗봇 만들 때도 동일한 원리를 적용한다. β 가 작으면 랜덤하게 다양한 대답이 나올 수 있도록 조절할 수 있다.

다음 문자를 예측할 때 β가 1인 standard softmax 함수를 사용하면, 계속해서 동일한 예측값이 나올 가능성이 높아진다. 따라서 β를 조절할 수 있는 softmax 함수 형태를 사용해 가끔은 다른 문자가 나오도록 한다.

Multinomial 분포 샘플링

해당 문자가 나올 확률에서, argmax로 뽑아낼 값을 다항 분포에서 샘플링한다. 그 다음 문자가 예컨대 3개라고 한다면, 3개 중 하나의 값을 다항 분포로부터 샘플링하는 것이다.

즉, a, b, c라는 문자 있고, 각 문자가 나올 확률로 모델이 예측한 값이 `[0.1, 0.25, 0.65]` 라고 하자. 기존의 방식대로라면 다음에 나올 문자로 c가 예측되겠지만, 이제 다항분포로부터 샘플링하게 되면, 다음에 나올 문자로 a가 뽑힐 확률이 `0.1`, b가 나올 확률이 `0.25`, c가 나올 확률이 `0.65`가 된다.

기존 방식대로 argmax하게 되면 다 똑같은 문자만 생성되어 나오기 때문에, 다양성이 떨어진다. 반면 multinomial sampling을 통해 argmax되는 인덱스를 조절해 준다.

> *질문* : vs. GAN?
>
> 시계열을 통해 과거에 어떤 문자가 나왔을 때 다음에 어떤 문자가 나올 확률이 높다고 학습하도록 한다. 시퀀스를 기반으로 한다. GAN은 확률 분포를 따라 만들어서 생성해 낸다는 점에서 다르다.

2. 실습: 코드 구현

2.1. 모듈 및 파일 로드

```
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LSTM, Activation
from tensorflow.keras.optimizers import RMSprop
import numpy as np
import random
import sys

path = '~'
with open(path) as f:
    text = f.read().lower() # 소문자
```

문자열을 인덱스로, 인덱스를 문자열로 매핑할 수 있도록 사전을 만든다.

```
characters = sorted(list(set(text))) # 모든 문자
char2indices = dict((c, i) for i, c in enumerate(characters))
indices2char = dict((i, c) for i, c in enumerate(characters))
```

문자열을 일정한 길이(`maxlen`)로 자른다.

```
maxlen = 40
step = 3
sentences = []
next_chars = []
for i in range(0, len(text) - maxlen, step):
    sentences.append(text[i: i + maxlen])
    next_chars.append(text[i + maxlen])
```

numpy array를 통해 3차원 시계열 데이터 형태로 문자열을 바꿔 준다. (기존에 했던 방식과 다른 면이 있으므로 코드 주의해서 살펴볼 것.)

```
X = np.zeros((len(sentences), maxlen, len(characters)), dtype=np.bool)
y = np.zeros((len(sentences), len(characters)), dtype=np.bool)
for i, sentence in enumerate(sentences):
    for t, char in enumerate(sentence):
        X[i, t, char2indices[char]] = 1
    y[i, char2indices[next_chars[i]]] = 1
```

모델을 구성한다. (Sequential 모델 이용)

```
model = Sequential()
model.add(LSTM(128, input_shape=(maxlen, len(characters))))
model.add(Dense(len(characters)))
model.add(Activation('softmax'))
model.compile(loss='categorical_crossentropy', optimizer=RMSprop(lr=0.01))
print (model.summary())
```

예측값을 인덱스로 매핑하는 함수를 만든다. 여기서 β 를 조절하고, multinomial sampling하는 과정이 포함된다.

- `metric` : 조절 변수 β.
- `preds` : 모델이 예측해 낸 softmax 값 array.

```
def pred_indices(preds, metric=1.0):
    preds = np.asarray(preds).astype('float64')
    preds = np.log(preds) / metric
    exp_preds = np.exp(preds)
    preds = exp_preds/np.sum(exp_preds)
    probs = np.random.multinomial(1, preds, 1)
    return np.argmax(probs)
```

한 번의 학습(`epoch`)을 진행한 뒤, 학습을 30번 반복(`iteration`)한다. 한 번의 학습 내에서 이후 400개의 문자를 예측한다. `Sine` 곡선 예측했듯, `generated`를 1문자씩 뒤로 `shift`해 가며 예측할 문자 범위를 뒤로 하나씩 밀어 준다.

β를 0.2, 0.7, 1.2로 바꾸어 가며 각각의 경우일 때 어떤 문자가 예측되는지를 보자.

```
for iteration in range(1, 30):
    print('-' * 40)
    print('Iteration', iteration)
    model.fit(X, y,batch_size=128,epochs=1)

    start_index = random.randint(0, len(text) - maxlen - 1)

    for diversity in [0.2, 0.7, 1.2]:
        print('\n----- diversity:', diversity)

        generated = ''
        sentence = text[start_index: start_index + maxlen]
        generated += sentence
        print('----- Generating with seed: "' + sentence + '"')
```

```
        sys.stdout.write(generated)

        for i in range(400):
            x = np.zeros((1, maxlen, len(characters)))
            for t, char in enumerate(sentence):
                x[0, t, char2indices[char]] = 1.

            preds = model.predict(x, verbose=0)[0]
            next_index = pred_indices(preds, diversity)
            pred_char = indices2char[next_index]

            generated += pred_char
            sentence = sentence[1:] + pred_char

            sys.stdout.write(pred_char)
            sys.stdout.flush()
        print("\nOne combination completed \n")
```

> *참고*
>
> 파일 입출력을 위해 `sys` 모듈을 사용했다.
>
> - `sys.stdout.write` : 콘솔에 출력.
> - `sys.stdout.flush` : 아무 것도 출력하지 않음.

첫 번째 iteration과 마지막 iteration의 결과를 비교하면 다음과 같다. 처음에는 제대로 모방하지 못하나, 학습을 반복하며 모델의 성능이 향상되면 필체가 잘 모방되는 것을 알 수 있다.

Iteration 1

```
Iteration 1
1515/1515 [==============================] - 12s 8ms/step - loss: 1.8849

----- diversity: 0.2
----- Generating with seed: "t him
    that he is lov'd of me; i foll"
t him
    that he is lov'd of me; i foll the sence shall we have son the prosess, and
some the compares and and shall we have me the sence my love him some some for
the sence of the prowness the confore the great shall we have be the great shall
the comparing of the sence and shall the call the prosent to see the sence the
ears and see the confore the growner the poor the sence and the send have the
prosent shall be the part shall the c
One combination completed


----- diversity: 0.7
----- Generating with seed: "t him
    that he is lov'd of me; i foll"
t him
    that he is lov'd of me; i foll shigh him since now sull to fill us;
    whit an a shall her have jack with your stind not in me bet will one, and
silf
```

```
    my leave, i his love with the art so me lave place wome high,
    the grain.
    which should the grage and strink your gave and fortrone him,
    the read sweet the eefe shill be the got that so now sore
    find not we ar suth of connortes of to or the eeving.
  cleopatra.
One combination completed


----- diversity: 1.2
----- Generating with seed: "t him
    that he is lov'd of me; i foll"
t him
    that he is lov'd of me; i foll' ming end.-


                                                                  exevinter
kiefing beto(!
   m) they befousiner will to hard, propydmw ove were him, of the rockes 'de
pell grtarish ting antof ill his gaod
    a we good fos nors ngien no l-swnir of peem
    which prryish nevem. mank would tru's hing in flids
  acect befor youn good hear ney live
    in there of belowner marself viui
One combination completed

----------------------------------------
```

Iteration 29

```
----------------------------------------
Iteration 29
1515/1515 [==============================] - 13s 8ms/step - loss: 1.2381

----- diversity: 0.2
----- Generating with seed: "    she carved thee for her seal, and me"
    she carved thee for her seal, and means me the master,
    the friends themselves the countess of the world leaf it will not in the
matter,
    and the countess of the world will be some commend.
  cleopatra. where i will not the countess of the strenger that i will death i
am
    the countess of soldiers, and shall she dispains.
    i will some of the world should have the matter,
    and the strenger the world for them,
    the co
One combination completed


----- diversity: 0.7
----- Generating with seed: "    she carved thee for her seal, and me"
    she carved thee for her seal, and mean
    them, and the greet antony, when they say now and in
    for them. arming, caesar is these strenger.
  cleopatra. hear love and ever second cheer. at my permission.
```

```
  cleopatra. come, may contents with her at to the recompinoned downran report
in the father,
    shall steep antony mings than at some for you!
    they will help learn of all after blow like the tranto of the days withed,
  an
One combination completed


----- diversity: 1.2
----- Generating with seed: "    she carved thee for her seal, and me"
    she carved thee for her seal, and men in it, and sualie you olion wcole;
    ywou and you, let me to the black all clown to kespomiel are rain done,
    ey o'er, buily he don sey both, as the ceuse,
  i would yeu mivight with thy false yondowned.
                    u meher sabidia,

acte ?jngu! them silncusate thenks be,
    woo by cortains! my good well, i am eniss, and be,
    and other, lack'n committillivent,
  for you airks wi
One combination completed
```

참고로, 코드에 오류가 있으면 아래와 같은 ~~망한~~ 결과가 나온다(^^).

```
epoch: 39
1515/1515 [==============================] - 8s 6ms/step - loss: 1.2051

       diversity: 0.2
Generating with Seed: 'y you shall not see me more; or if,
    '
y you shall not see me more; or if,

 cwbbccbcwbbcbbbbcbbbbcbbmbbmbbfbbcbbbbcfbcbbbbcbbbbbbbbbcbbbbcbbwbbbbwcbbcbcbcb
cbbbbbbbbwwbbbbfbbwbbbwccbbwbbcbcbbbcbbbbcbbbbbbwcbcbbbcbpbbmcbbbfbpcbbbbbbbcbcc
bbbfbbccbbcbbbcbbwbwcbpbbvbcbcbbbbbccbbccbffbbwcfbbbccbfbbbmbbbwbbbbbcfcbcbbwcbc
cbwbbbbcfbbfbbbbbbbbbcwbbbbcbbbfbbbfcbcbbccbcwbbbccbmbbwbbbcbbbwcbbbbbbbbgcbcbbb
bfbccbbbbbcwbfbbbbbbbcbbbbpbcbbbbbpbbbcbcmcfbbbbbbbbbbbbbbbbbcccbbfpbbbwfbpwcbbm
b

       diversity: 0.7
Generating with Seed: 'y you shall not see me more; or if,
    '
y you shall not see me more; or if,

 bwfccpbgskfwqafwbfcvfvbbumwmpctcbpbbpcfcfcvwfybbbwgwppbtffkpfgvwwpfbcdbwacwnwpt
grtwbbcbbbfbccgpkgwsgvhwfubwbi'bgcjwbpsgiwmmbfqfqkfvcvwpgicgcwbwcciwmabfmbblbpfw
kfvwcwpkbbccbcfqfkbwmwgbpcawsbpcbgfflmfsvmstmwbnpwwmfwcfmcwmgwvptwcmcfbakvfbwgmf
bbwwbcmfbbccmfbbccsgwgmmvbbtrfctcwmbfbmmbbbwnfacbwcvbbmqfbcmfbpbwvmccmctcwibcwbb
cwbmbffbfwplbbmuqbcfwcfpbwfmswfabsbfbblpgbpnbivwbgvbbqqcunrmvjpiwfffcfbcwppcucfc
c
```

```
       diversity: 1.2
Generating with Seed: 'y you shall not see me more; or if,
    '
y you shall not see me more; or if,

 bcwmtpbpw'wvbrsskociftllntbbatbvkmvkfispbwmmqhvvbwcmdwbpybuycgfbkmckccqvicjwcpr
wbuvwmqvbwgbgrjubptwbmtafbpccjmlwsfkcctfomgumnvobmwcrbbfgcbblkjbcefamwqbgkcckmvf
fnxkanmbnafwmmkpspbwpqbfmgsccgecfbfqdpvccbntkniucphfudhuwpcvvwqstcufcvwwvamufpfi
bfffqmkbdmgbwwwgw'wqccnmbcwoyg.b?
lqwwnnfbwwbukgvwwbmvvccftgvfrcguuvwfbpfvwdbtfpbwcnvgwpmfkcmipwqiwsgkgmcrdctwwfmu
blcnafbgabwpgwgpocpmvpbbsmlfmppnrfawbtcpnq.vpr-q
```