Deep Learning을 활용한 Language Sequence 처리

DEEP LEARNING AND NATURAL LANGUAGE PROCESSING

10

APPLICATION

ARTIFICIAL INTELLIGENCE

Notice

THINKING

생각해보기

✓ Deep Learning을 활용한
Language Sequence 처리는 어떻게 할까요?

학습목표

GOALS

1 Seq2Seq모델을 이해하고 설명할 수 있다.

2 Seq2Seq모델이 사용되는 곳에 대해 이해하고
설명할 수 있다.

3 Encoder-Decoder 모델을 이해하고 설명할 수 있다.

4 Seq2Seq 모델의 작동 범위를 이해하고 설명할 수 있다.

5 실제 데이터를 통해 Seq2Seq 모델의 동작을 이해하고
설명할 수 있다.

CONTENTS

1 Sequence to Sequence

2 Encoder-decoder

3 Implementation & dataset

4 Encoder - Decoder inputs

5 Training

6 실습

학습내용

APPLICATION

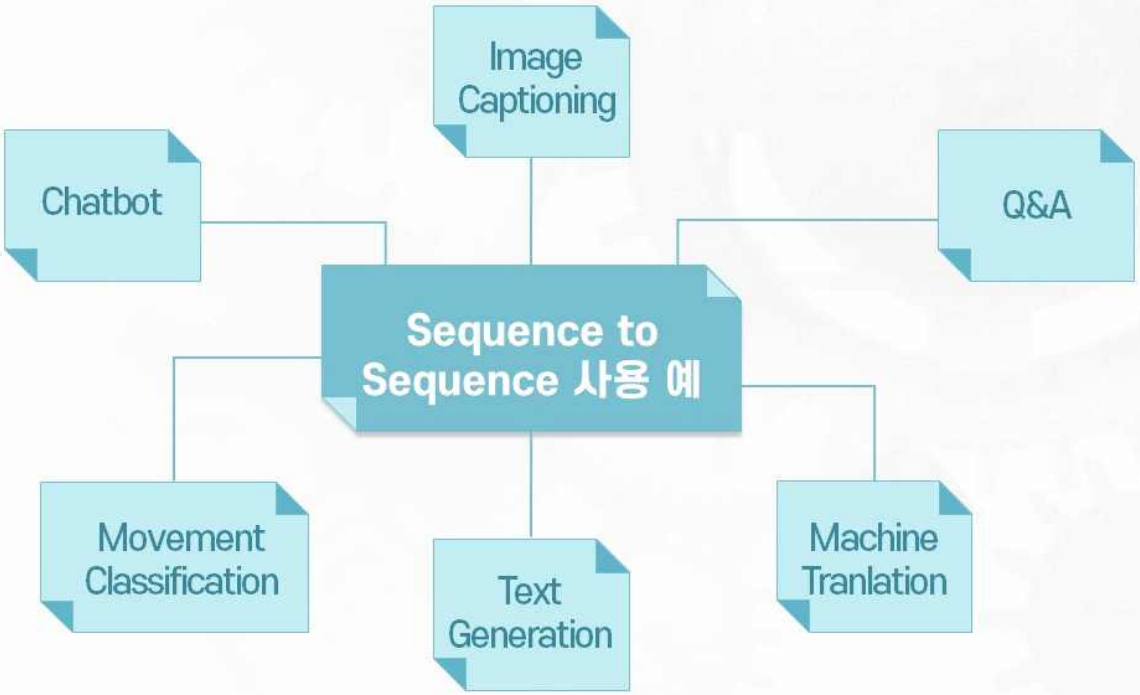
01 Seq2Seq 모델 이해

01 Sequence to Sequence란?

연속된 형태의 Data을 입력 받아

연속된 형태의 Data을 **출력**하는 모델

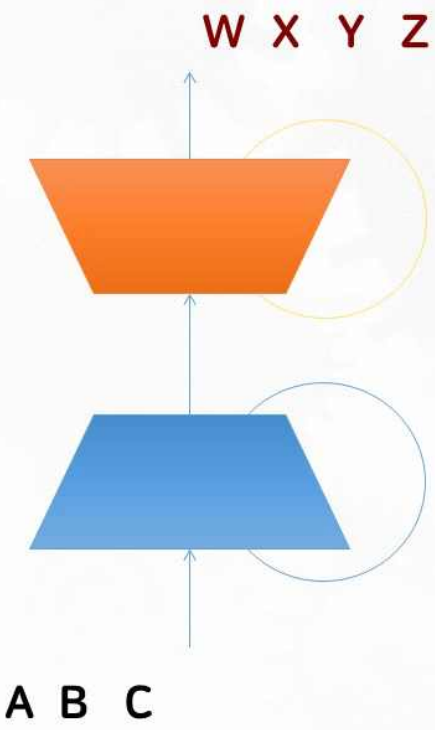
02 Sequence to Sequence 사용 예



DEEP LEARNING AND NATURAL LANGUAGE PROCESSING

03 Encoder-decoder

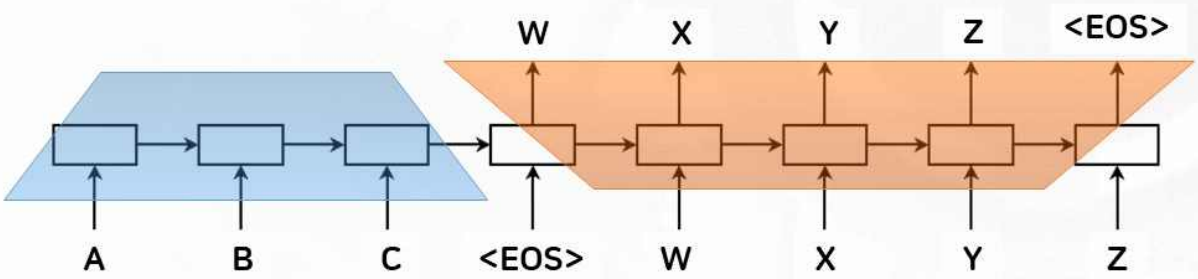
model



03 Encoder-decoder model

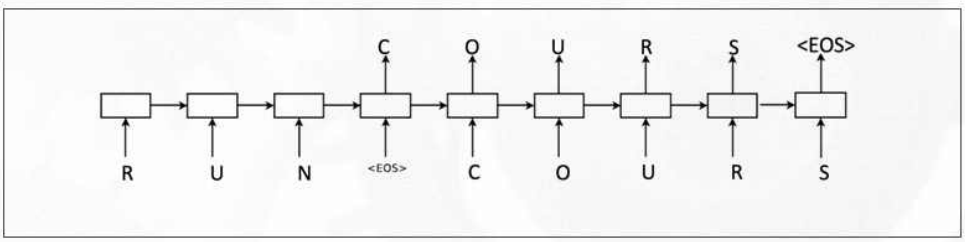
Input ABC

Output WXYZ

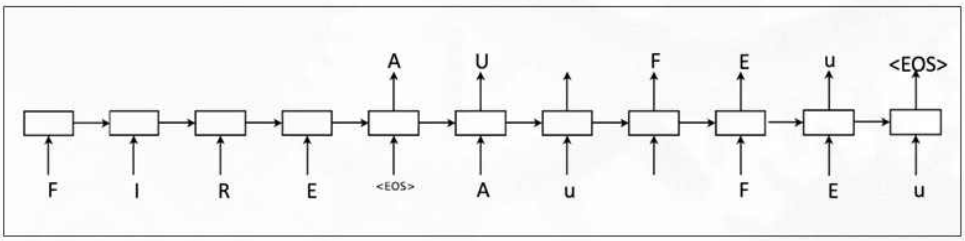


04 Example

Input (english) : RUN
Output (French) : COURS



Input (english) : FIRE
Output (French) : AU REu



Eng-French dataset

- 영어 총 문자 수 : 90
- 프랑스 총 문자 수 : 113

English : Go. French : Va !
English : Hi. French : Salut !
English : Hi. French : Salut.
English : Run! French : Cours !
English : Run! French : Courez !
English : Who? French : Qui ?
English : Wow! French : Ça alors ! English : Fire! French : Au feu !
English : Help! French : À l'aide ! English : Jump. French : Saute.

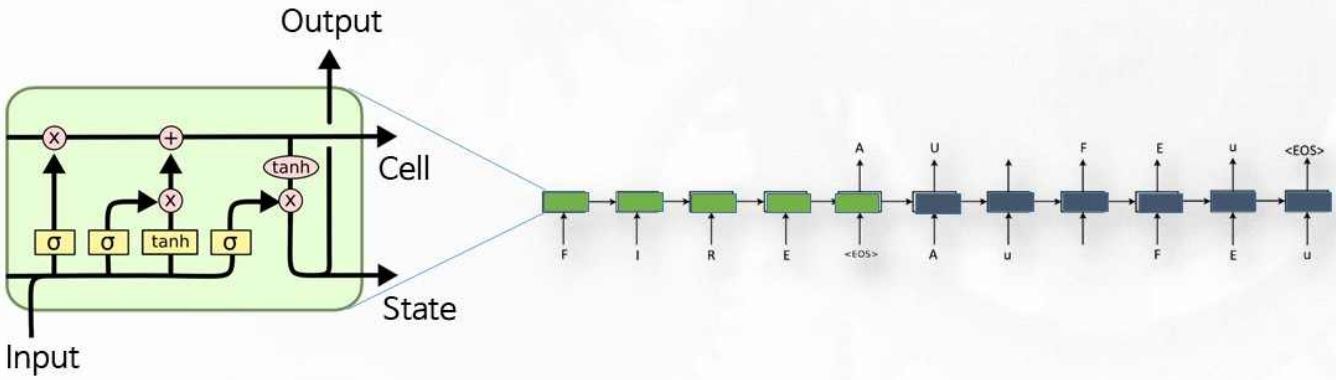
APPLICATION

02 Seq2Seq 모델 구축

01 Implementation & dataset

DEEP LEARNING AND NATURAL LANGUAGE PROCESSING

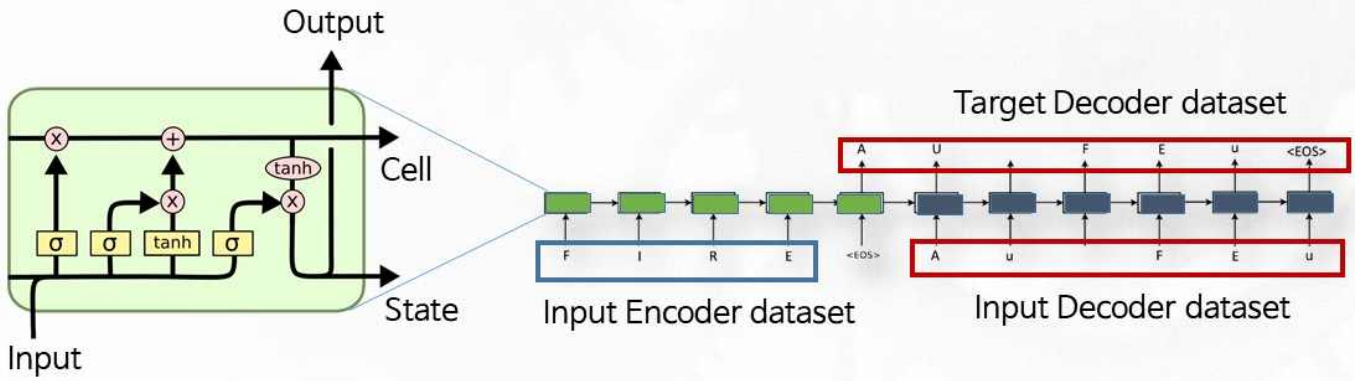
Implementation



Encoder Cell : LSTM
Decoder Cell : LSTM

출처: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

dataset



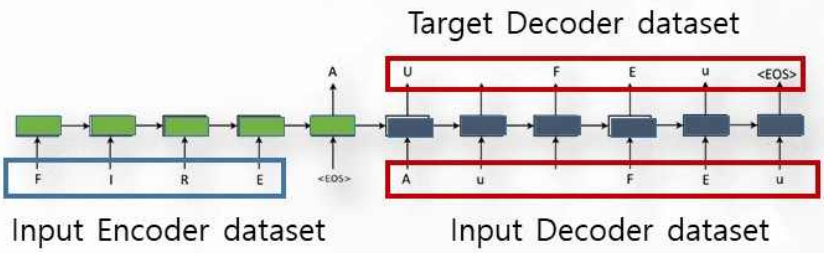
Encoder Cell : LSTM
Decoder Cell : LSTM

출처: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

01 Implementation & dataset

DEEP LEARNING AND NATURAL LANGUAGE PROCESSING

dataset



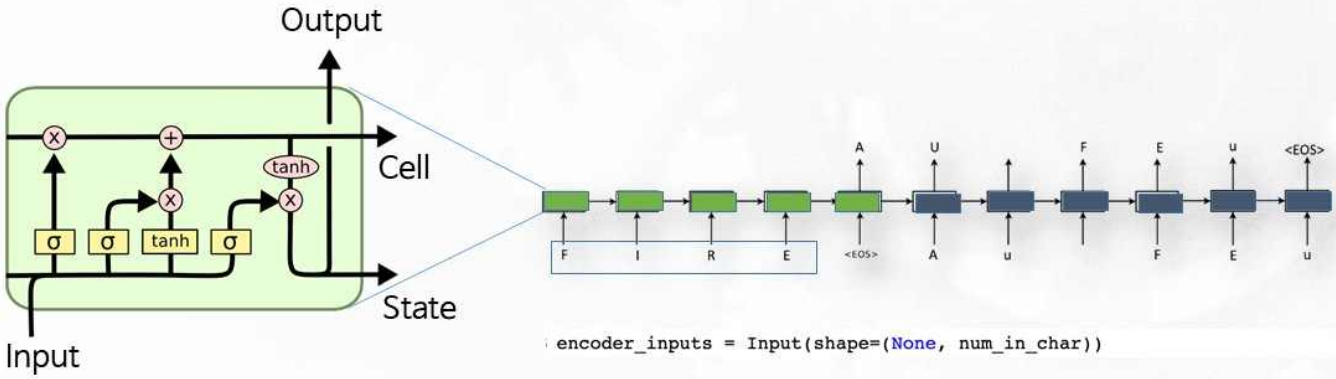
Input Encoder dataset :	F	I	R	E	⟨EOS⟩

Input Decoder dataset :	A	U		F	E	u

Target Decoder dataset	U		F	E	u	⟨eos⟩

출처: 퍼블릭에이아이(www.publicai.co.kr)

Encoder - input shape




```
encoder_inputs = Input(shape=(None, num_in_char))
```

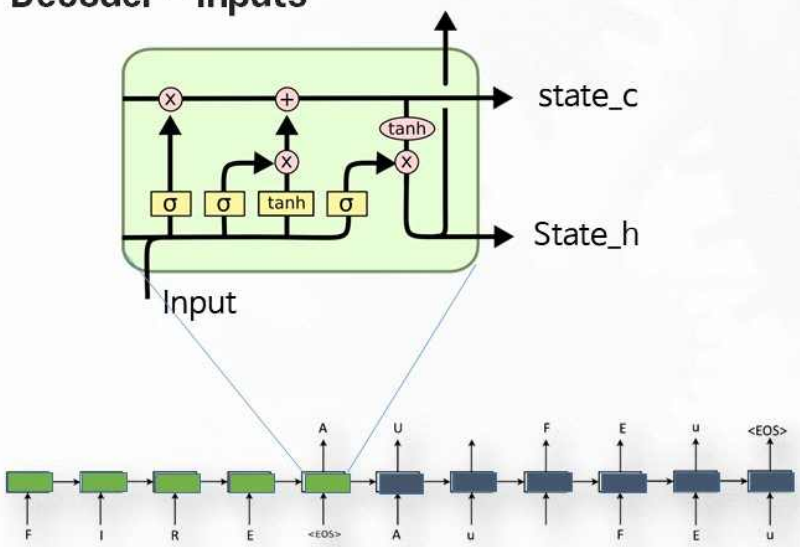
Input shape : (batch size, None, num_English_char)

출처: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

02 Encoder - Decoder inputs

DEEP LEARNING AND NATURAL LANGUAGE PROCESSING

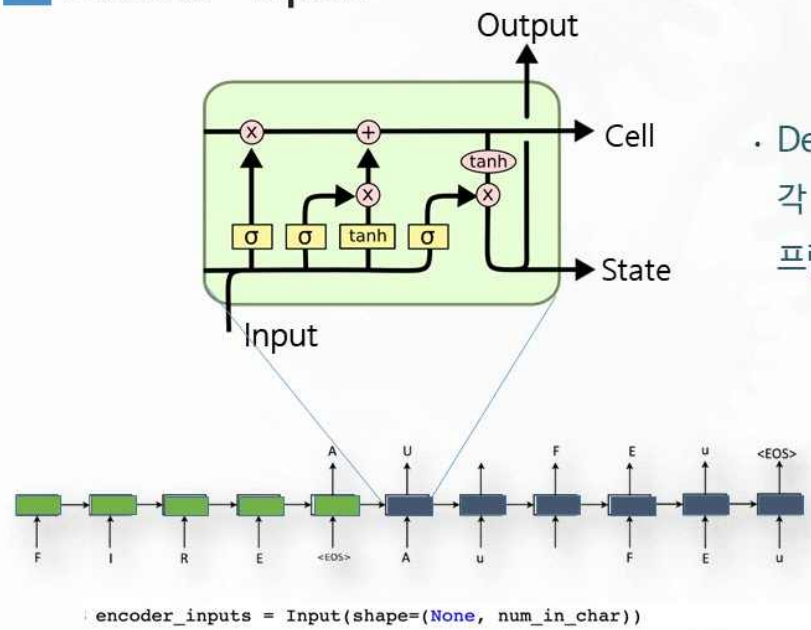
Decoder - inputs




```
# LSTM 모델을 사용합니다.
encoder = LSTM(latent_dim, return_state=True)
encoder_outputs, state_h, state_c = encoder(encoder_inputs)
```

출처: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Decoder - inputs



- Decoder 에서도 timestep 은 가변적이며 각 timestep 마다 입력 또는 출력의 vector 크기는 프랑스 문자열의 모든 문자 종류와 같음

```
encoder_inputs = Input(shape=(None, num_in_char))
```

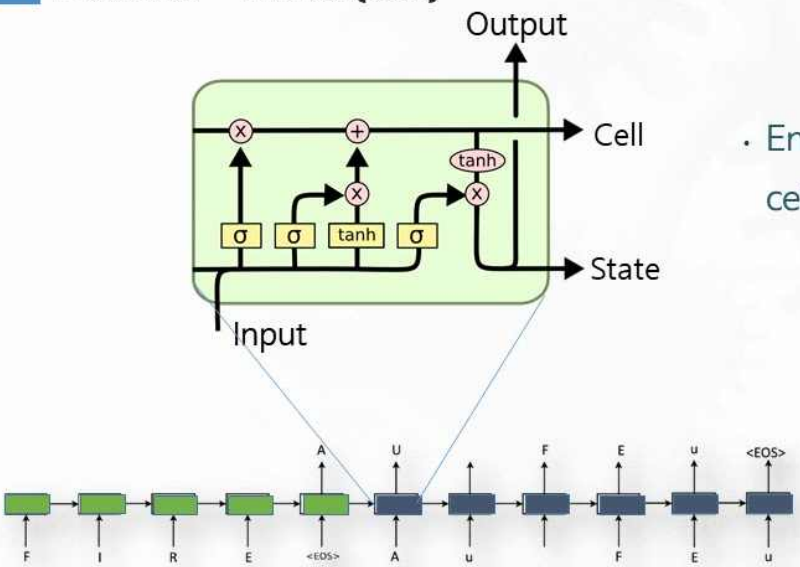
Input shape : (batch size, None, num_English_char)

출처: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

02 Encoder - Decoder inputs

DEEP LEARNING AND NATURAL LANGUAGE PROCESSING

Decoder - model(cell)

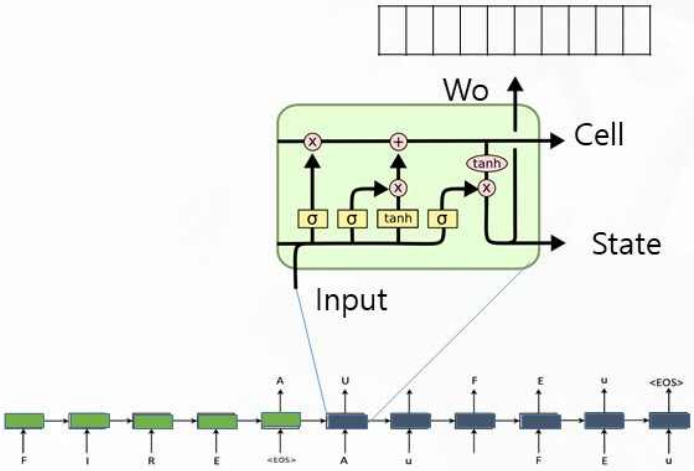


- Encoder 에서 마지막 hidden state weights와 cell_weights 을 재사용함

```
decoder_lstm = LSTM(latent_dim, return_sequences=True, return_state=True)
decoder_outputs, _, _ = decoder_lstm(decoder_inputs,
                                     initial_state=encoder_states)
```

출처: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Decoder- model(output)



- Encoder 달리 Decoder는 모든 timestep Output을 사용함
- Classification 출력을 하기 위해 Dense Layer-softmax을 적용함

```
decoder_lstm = LSTM(latent_dim, return_sequences=True, return_state=True)
decoder_outputs, _, _ = decoder_lstm(decoder_inputs,
                                     initial_state=encoder_states)
decoder_dense = Dense(num_out_char, activation='softmax')
decoder_outputs = decoder_dense(decoder_outputs)
```

출처: http://colah.github.io/posts/2015-08-Understanding-LSTMs/


```
# Run training
model.compile(optimizer='rmsprop', loss='categorical_crossentropy')
model.fit([encoder_input_data, decoder_input_data], decoder_target_data,
          batch_size=batch_size,
          epochs=epochs,
          validation_split=0.2)
# Save model
model.save('s2s.h5')
```

APPLICATION

03 실습

SUMMARY

학습정리

- Seq2Seq의 등장 배경
- Seq2Seq의 기본 아이디어
- Encoder-Decoder 모델
- Seq2Seq의 구조
- Seq2Seq의 input
- Seq2Seq 실제 문제에서의 활용

EXPANSION

확장하기

1. Seq2Seq 모델이 등장하게 된 배경은 무엇일까요?
2. Seq2Seq 알고리즘의 기본 아이디어는 무엇일까요?
3. Seq2Seq 알고리즘의 기본 구조는 어떻게 구성될까요?
4. Encoder-Decoder 모델은 무엇일까요?
5. Seq2Seq 알고리즘은 실제로 어떻게 적용할 수 있을까요?

참고 문헌

REFERENCE

◆ 참고 사이트
- 용어들에 대한 정의 : https://ko.wikipedia.org/wiki.
- 퍼블릭에이아이(www.publicai.co.kr)
- http://colab.github.io/posts/2015-08-Understanding-LSTMs/

◆ 참고 논문
- I Sutskever. Sequence to Sequence Learning with Neural Networks, 2014

서체 출처 : 에스코어드림체-㈜에스코어, 나눔글꼴체-㈜네이버