

## Lecture11: ConvNets for NLP

### CNNs

- 2d example →
- Yellow color and red numbers show filter (=kernel) weights
- Green shows input
- Pink shows output

1 <sub>x1</sub>	1 <sub>x0</sub>	1 <sub>x1</sub>	0	0
0 <sub>x0</sub>	1 <sub>x1</sub>	1 <sub>x0</sub>	1	0
0 <sub>x1</sub>	0 <sub>x0</sub>	1 <sub>x1</sub>	1	1
0	0	1	1	0
0	1	1	0	0

Image

4		

Convolved Feature

From Stanford UFLDL wiki

CNN은 동일한 weight를 슬라이드하면서 행렬에 곱하는 것이 핵심이다. 이미지에선 똑같은 이작은 요소라도 이미지 내의 위치, 회전, 밝기 등에 따라 다르게 표현된다. 그리고 하나의 필터는 하나의 특징을 잡아내는 역할을 한다. 어떤 필터는 대각선을, 어떤 필터는 밝은 색을, 어떤 필터는 원형을 잡아내는 식이다. 그래서 하나의 이미지에 하나의 필터가 통과하여 만들어지는 행렬은 그 이미지가 해당 특징을 어느 정도로 가지고 있는 지에 대한 일종의 분포라고 할 수 있다. 하나의 이미지에 다수의 필터를 적용하게 된다. 하나의 필터는 하나의 특징 맵을 만들게 되고, 여러 개의 특징 맵이 묶여서 새로운 채널을 이루게 된다. 즉, 하나의 이미지에 여러 필터가 통과되어 나온 특징 맵들은, 해당 이미지가 가지고 있는 특징들의 표현이다.

∅	0.0	0.0	0.0	0.0
tentative	0.2	0.1	-0.3	0.4
deal	0.5	0.2	-0.3	-0.1
reached	-0.1	-0.3	-0.2	0.4
to	0.3	-0.3	0.1	0.1
keep	0.2	-0.3	0.4	0.2
government	0.1	0.2	-0.1	-0.1
open	-0.4	-0.4	0.2	0.3
∅	0.0	0.0	0.0	0.0

∅,t,d	-0.6	0.2	1.4
t,d,r	-1.0	1.6	-1.0
d,r,t	-0.5	-0.1	0.8
r,t,k	-3.6	0.3	0.3
t,k,g	-0.2	0.1	1.2
k,g,o	0.3	0.6	0.9
g,o,∅	-0.5	-0.9	0.1
ave p	-0.87	0.26	0.53

Apply 3 filters of size 3

3	1	2	-3
-1	2	1	-3
1	1	-1	1

1	0	0	1
1	0	-1	-1
0	1	0	1

1	-1	2	-1
1	0	-1	3
0	2	2	1

## Pooling

뽑힌 feature를 summarize하고 싶을 때 주로 사용되는 기법은 Max pooling 기법이 있다. Max pooling 기법은 각 Filter 중에서 가장 큰 값을 뽑아주어 하나의 vector로 만들어준다. K-max pooling은 Max pooling과 달리 하나의 feature만 뽑는 것이 아니라 가장 큰 k개의 feature를 뽑는 기법이다. 이때 주의해야 할 점이 출력된 값의 크기대로 정렬해서 적어주는 것이 아니라 값이 있는 그 위치를 보존해 적어주어야 한다. 위치를 보존하게 되면 그 단어의 위치에 대한 정보까지 반영이 된다고 한다. Average pooling은 말 그대로 Filter의 평균값을 적어주는 기법이다. 하지만 NLP에서는 각 문장에서의 단어의 중요도나 위치값이 필요하기 때문에 이방법은 잘 사용하지 않는다. 예시에서는 Stride를 1로 convolve를 했다. Stride를 1이 아닌 다른 값으로 하면 어떻게 될까? Stride를 2로 크게두어 진행하게 되면 상대적으로 연산량이 많이 줄어들고, 출력되는 Sequence의 길이 역시 줄어들게 된다. 하지만 NLP에서는 단어들의 정보를 중요시(?)하기 때문에 이런 방식을 잘 사용하지 않는다. local max pooling은 Vision에서는 엄청 많이 사용되는 기법 중 하나이다. stride를 2 두고 해당 영역의 Local max pooling을 통해서 값을 산출하고, 계산하는 방식이다.

## Simple CNN for sentence classification

## Deep CNN for sentence classification

### CNN for Translation

최초의 기계번역 시도는 seq2seq이 아니었다고 한다. 실제로는 2013년에 발표된 CNN을 인코더로 하는 모델이라고 한다. 이 모델은 문장을 Conv 레이어를 여러겹 통과시켜 sentence encoding을 하고, 이를 RNN류 모델을 디코더로 하여 번역을 시도했다고 한다.

### Very Deep Convolutional Networks for Text Classification

2017년은 컴퓨터 비전 분야에 있어서 변화의 시기였다. Residual Connection의 발견으로 엄청 깊은 네트워크를 설계할 수 있었기 때문이다. NLP에서는 그에 비해 3, 4겹의 LSTM을 쌓는 수준이었으니, 당시 연구자들은 좀 다른 시도를 통해 깊은 네트워크를 쌓고자 했다고 한다.