



빅데이터와 인공지능의 이해

08.딥러닝 알고리즘

조창제

강의자료

2025.03.

목차

CONTENTS



I

인공신경망의 배경과 기초 레이어

개요

II

인공신경망의 기초부터

인공신경망

III

합성곱신경망의 기초 레이어

합성곱신경망



목차

CONTENTS



IV

순환신경망의 기초 레이어

순환신경망

V

자료의 압축과 복원

모델구조

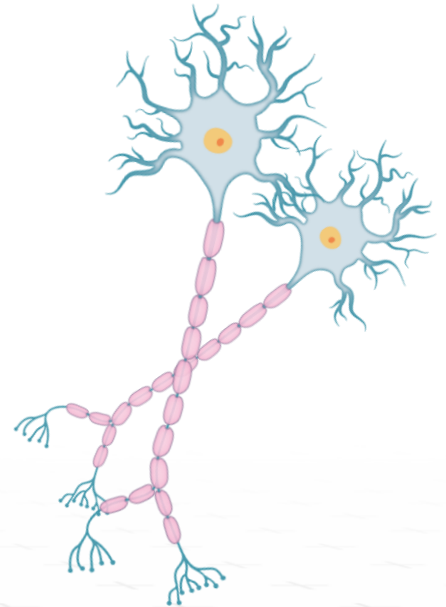
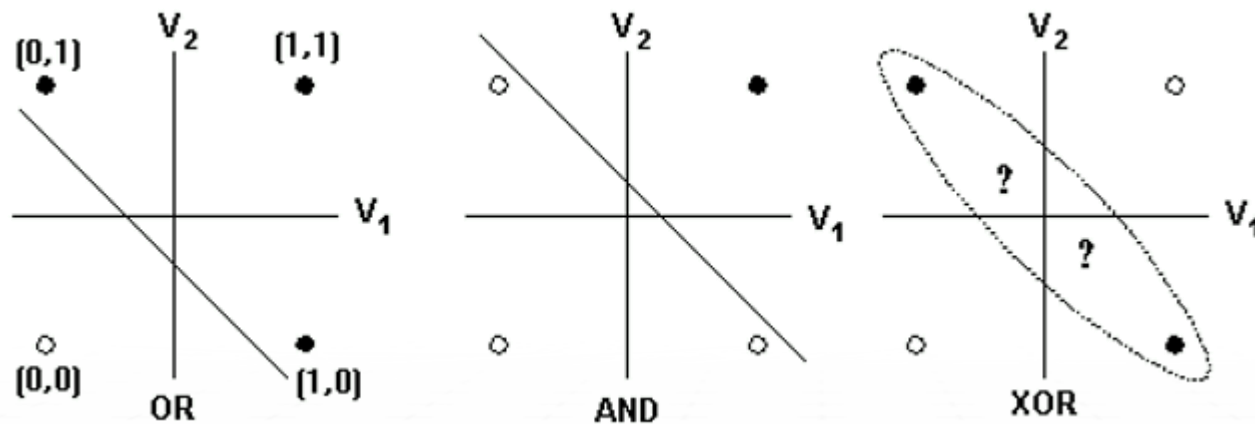




1. 배경

1) 퍼셉트론(Perceptron, 1958)

- ① 인간의 신경전달을 담당하는 매개체인 **뉴런의 신경 전달 원리를 모방**하여 설계된 모델
- ② 활성화함수: 뉴런의 역치(특정 수치 이상에서만 반응)를 모방한 함수
- ③ XOR 문제
 - XOR 문제를 해결하지 못해 침체기가 발생
 - XOR문제는 아래 그림에서 검은 점과 흰 점을 구분하는 선을 찾는 문제





01. 발전과정

1. 배경

2) 다층 퍼셉트론(Multi layer perceptron, 1986)

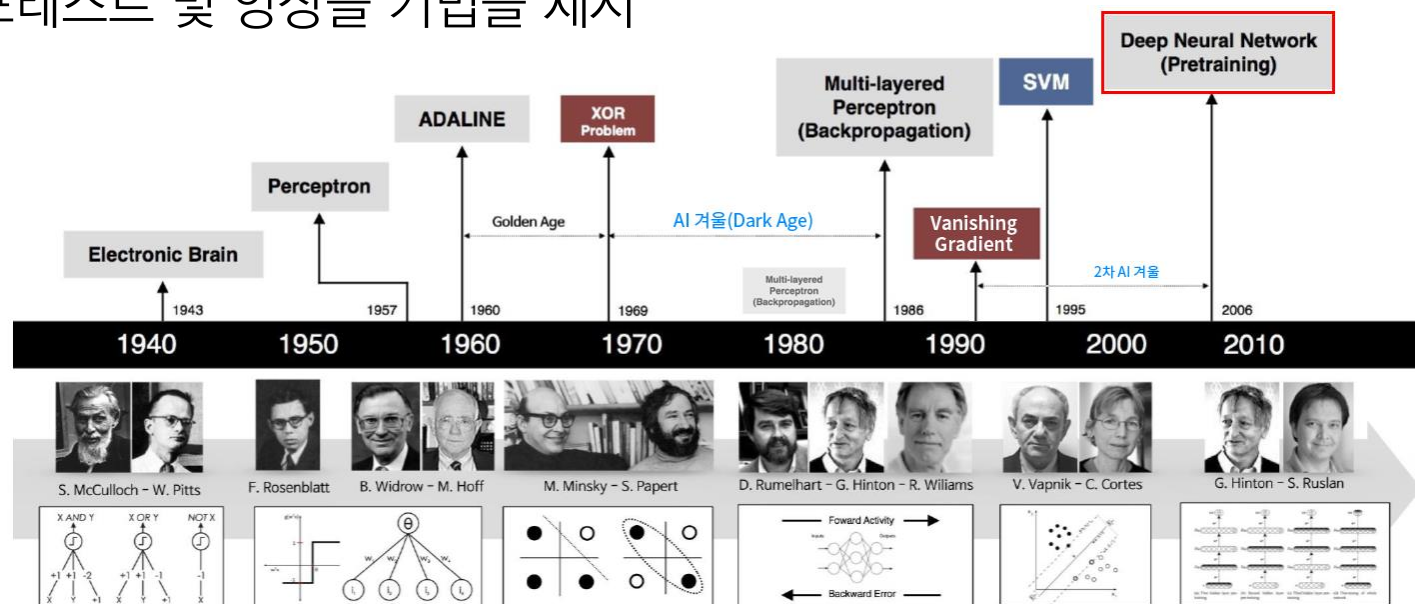
① XOR문제 해결

- 은닉층을 2개 이상 쌓음으로써 XOR문제를 해결

3) 다양한 머신러닝 기법 등장

① SVM(1990), 의사결정나무, 랜덤 포레스트 및 앙상블 기법들 제시

② 하드웨어 한계가 존재





01. 발전과정

1. 배경

4) 합성곱 신경망(CNN, 2012~)

① AlexNet(2012)

- 이미지 처리 문제에서 딥러닝 방법을 적용

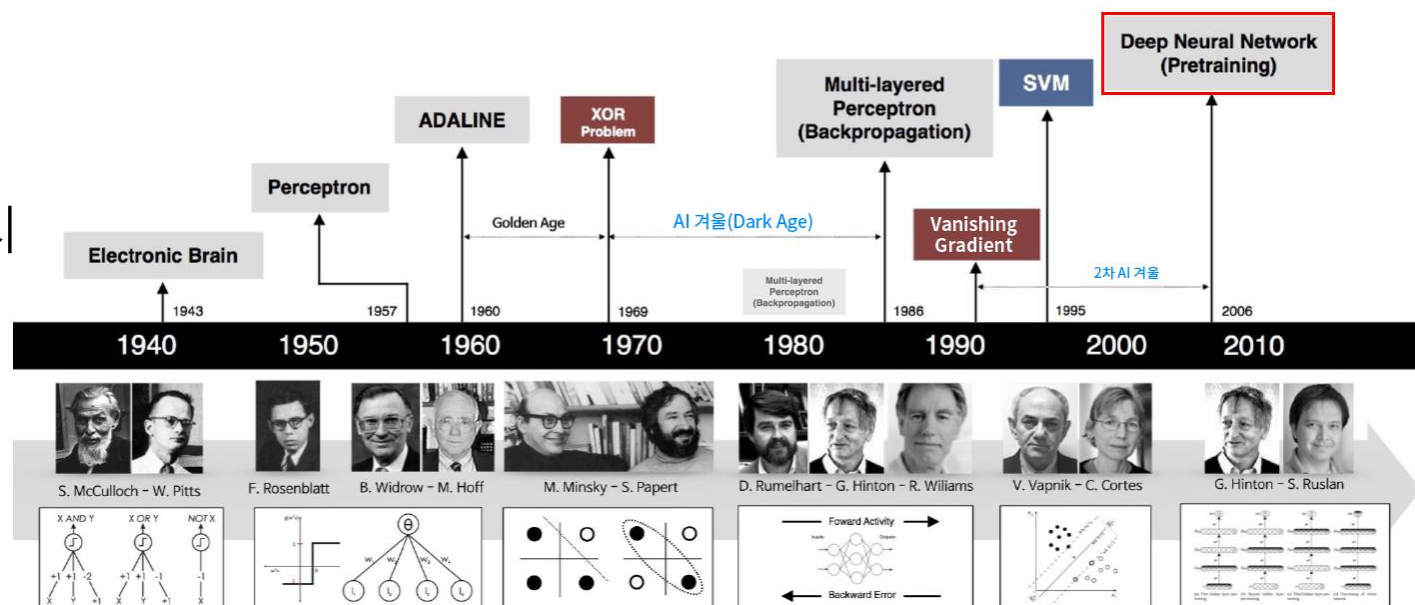
5) GAN(2014)

① 이미지 생성 모델 개발

② 초해상화 및 도메인 전환 연구 활용

6) RCNN(2014), YOLO(2016~)

① AI만으로 객체를 인식하는 모델 제시





01. 발전과정

1. 배경

7) Attention, Transformer(2017)

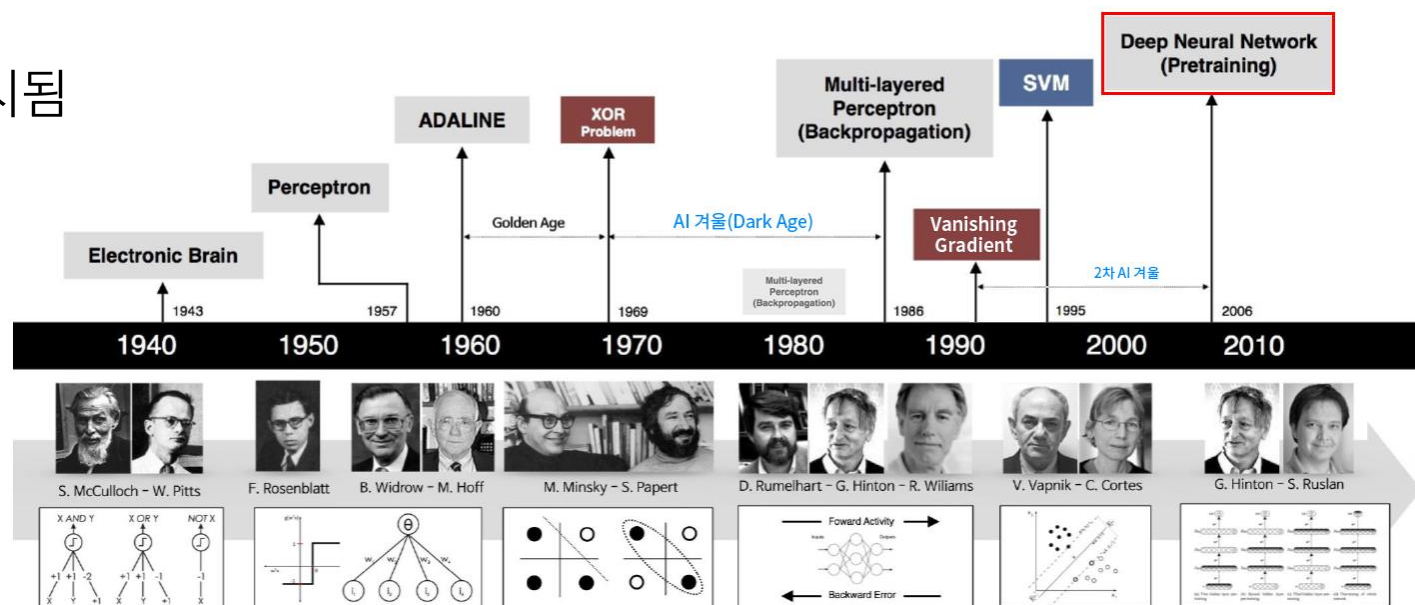
- ① 자연어 처리 및 생성 모델의 발전

8) Diffusion(2022)

- ① 이미지 생성 모델의 발전

9) Whisper(2022)

- ① OPENAI사의 음성 인식 모델이 제시됨





1.정의

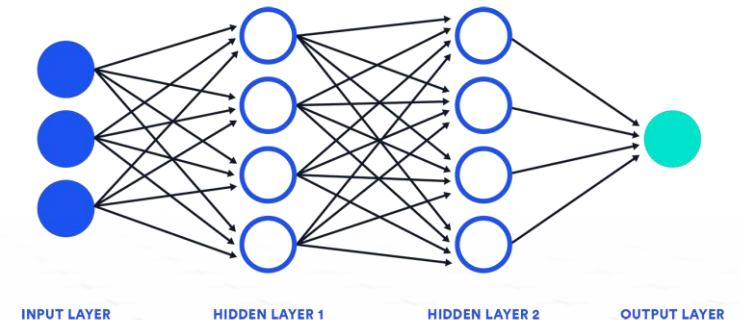
1) 인공신경망(Artificial neural network)

① 인간의 뇌에서 영감을 받아 만들어진 모델

- 입력층, 은닉층, 출력층으로 구성
- 딥러닝: ANN 중 은닉층이 여러 개로 구성된 것을 의미
- 은닉층은 Dense, CNN, RNN과 같은 레이어를 의미하며, 가중치를 포함하고 있음
- $Y = XW + b$ 의 형태
- 가중치는 X의 shape이 3이고, Y의 shape이 64이면 W의 shape은 (3, 64)이고 행렬곱을 통해 계산

2) MLP(Multi Layer Perceptron)

① 완전연결층(Fully connected layer, FC layer)로 구성된 신경망



1.정의

1) 드롭아웃(Dropout)

① 훈련 중에 매 배치마다 뉴런을 무작위로 비활성화하여 과적합을 방지하는 기법

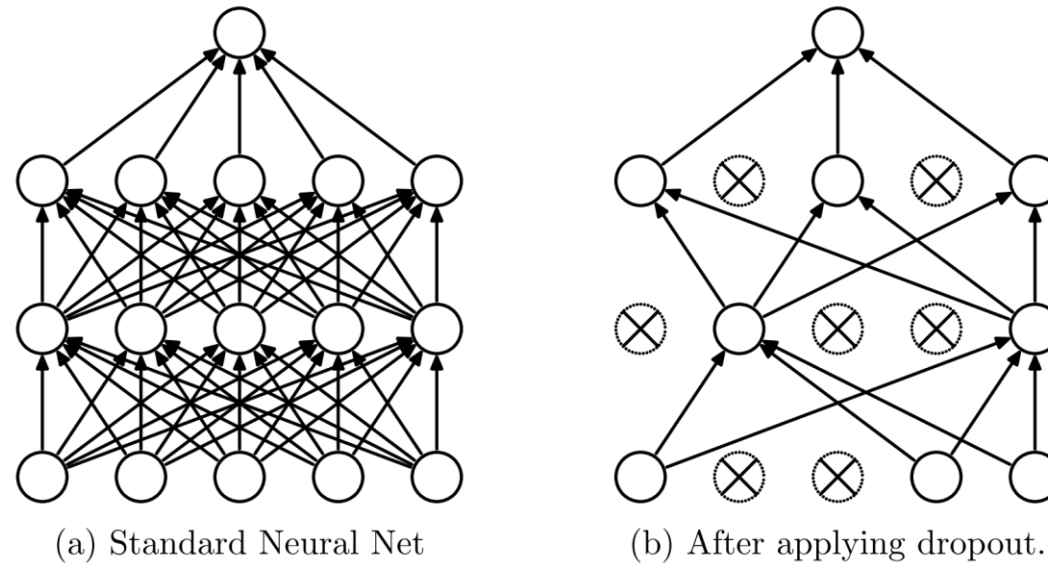
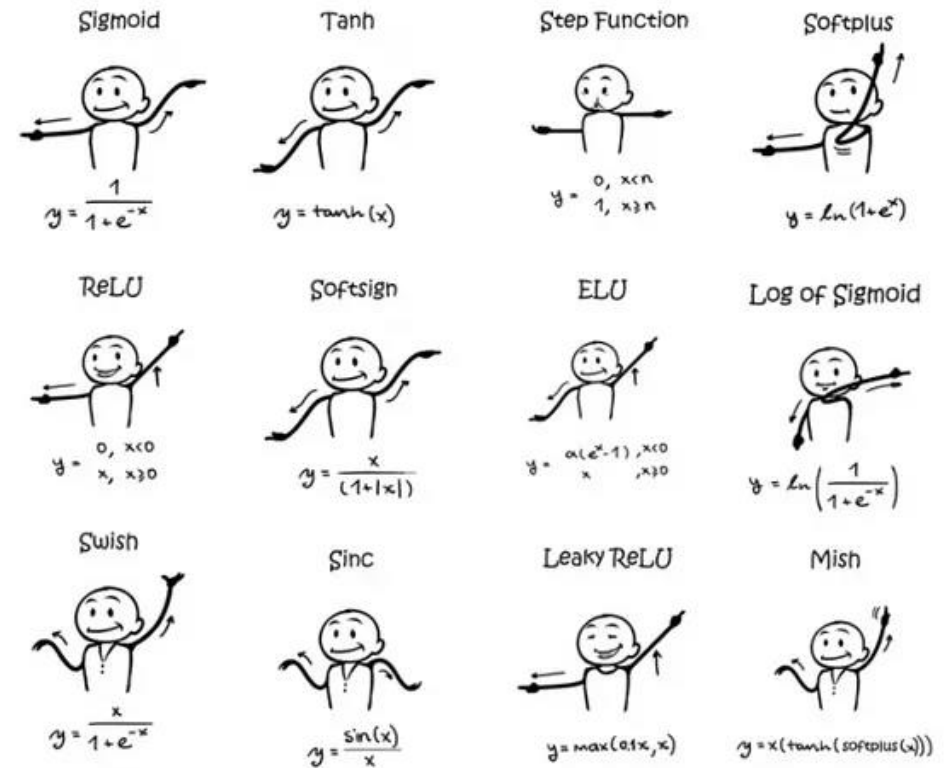


Figure 1: Dropout Neural Net Model. **Left:** A standard neural net with 2 hidden layers. **Right:** An example of a thinned net produced by applying dropout to the network on the left. Crossed units have been dropped.

1.정의

1) 활성화함수(Activation function)

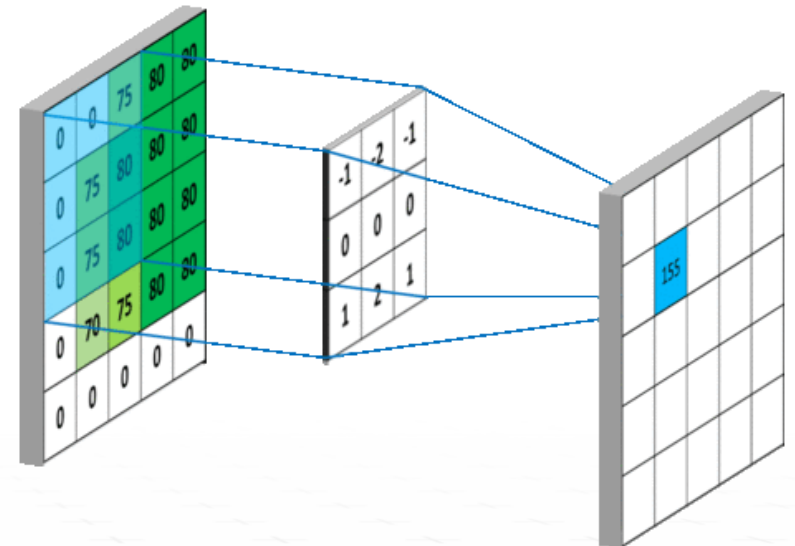
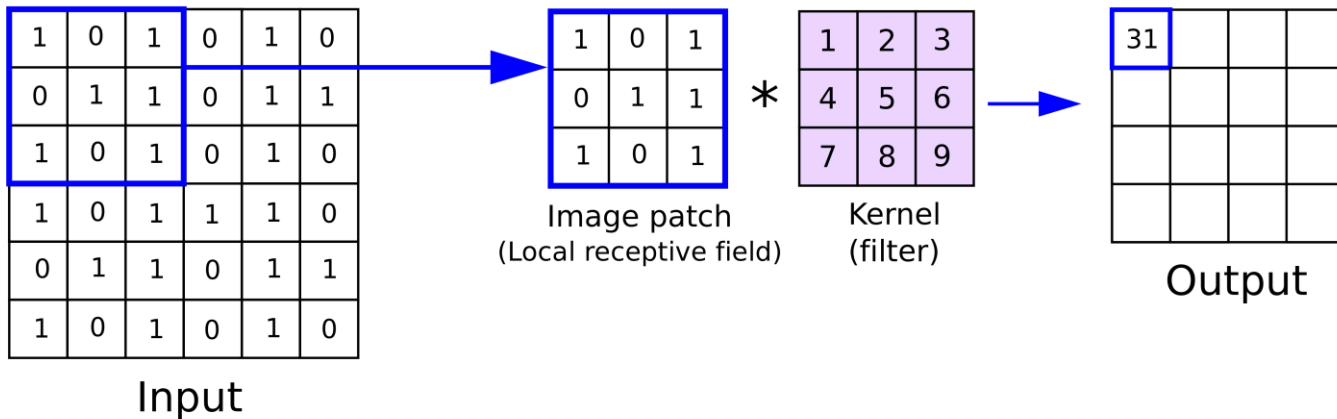
- ① 뉴런이 활성화 될지 말지를 결정
- ② 레이어 중간에서는 주로 ReLU나 Leaky ReLU를 사용
- ③ 최종 레이어 결정
 - 이진분류 문제: Sigmoid
 - 다지분류 문제: Softmax
 - 회귀문제: Linear



1.정의

1) 합성곱 층(Convolutional layer)

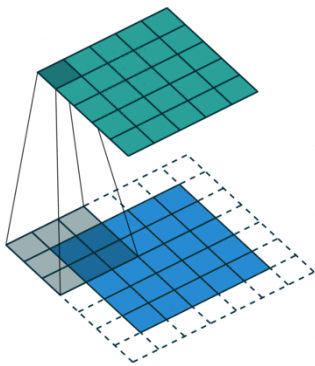
- ① 필터(kernel)을 사용하여 입력 데이터를 슬라이딩하면서 합성곱 연산을 수행하는 레이어
- ② 합성곱 연산: 커널값과 커널크기의 입력값을 곱한 후 더하는 연산
 - kernel size: 풀링에서 이동하는 윈도우 크기
 - stride: 윈도우가 움직일 때 이동하는 간격
 - padding: 입력 데이터의 경계에 추가적인 영역을 생성해주는 방법
 - dilation: 커널 간의 간격을 의미



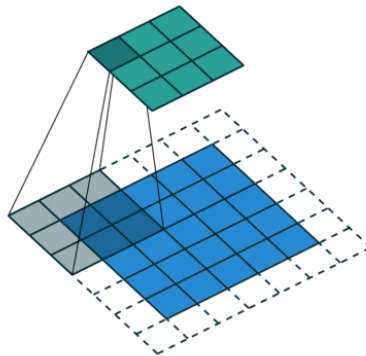
1.정의

1) 합성곱 층(Convolutional layer)

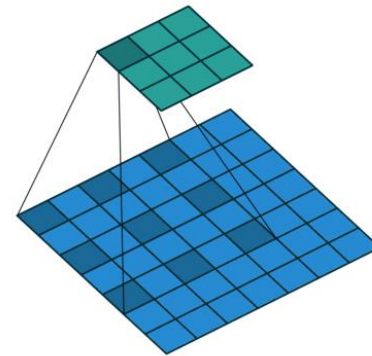
- kernel size: 풀링에서 이동하는 윈도우 크기
- stride: 윈도우가 움직일 때 이동하는 간격
- padding: 입력 데이터의 경계에 추가적인 영역을 생성해주는 방법
- dilation: 커널 간의 간격을 의미



kernel=3
stride =1
padding=1
dilated=1



kernel=3
stride =2
padding=1
dilated=1



kernel=3
stride=1
padding=0
dilated =2

1.정의

2) Depthwise separable convolutional layer

① Depthwise convolutional layer

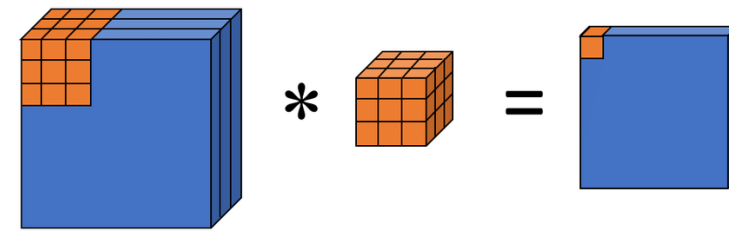
- 채널별로 연산을 따로 수행
- 계산의 효율성을 높이기 위해 사용
- Conv2d기준: $(K, K, C_{in}) = C_{out} * C_{in} * K * K$ 개 파라미터
- Depthwise기준: $(K, K, 1) = C_{in} * K * K$ 개 파라미터

② Pointwise convolutional layer

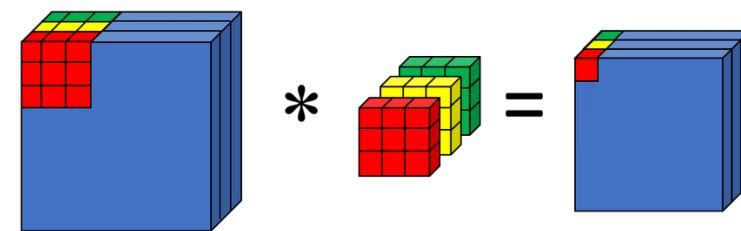
- kernel 크기가 1x1인 Conv layer를 의미
- Pointwise 기준: $(1, 1, C_{in}) = C_{out} * 1 * 1$ 개 파라미터

③ 두 레이어 합계

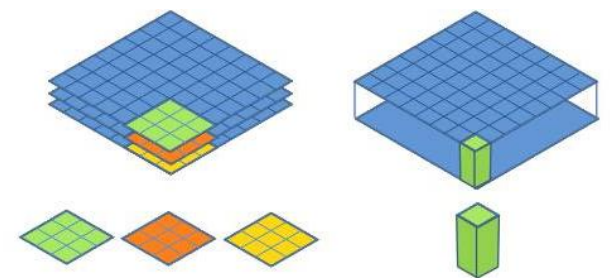
- $C_{in} * K * K + C_{out}$
- Conv2d 대비 약 $K * K$ 배 만큼 감소



Conventional Convolution



Depthwise Convolution



Depthwise Convolutional Filters

Pointwise Convolutional Filters



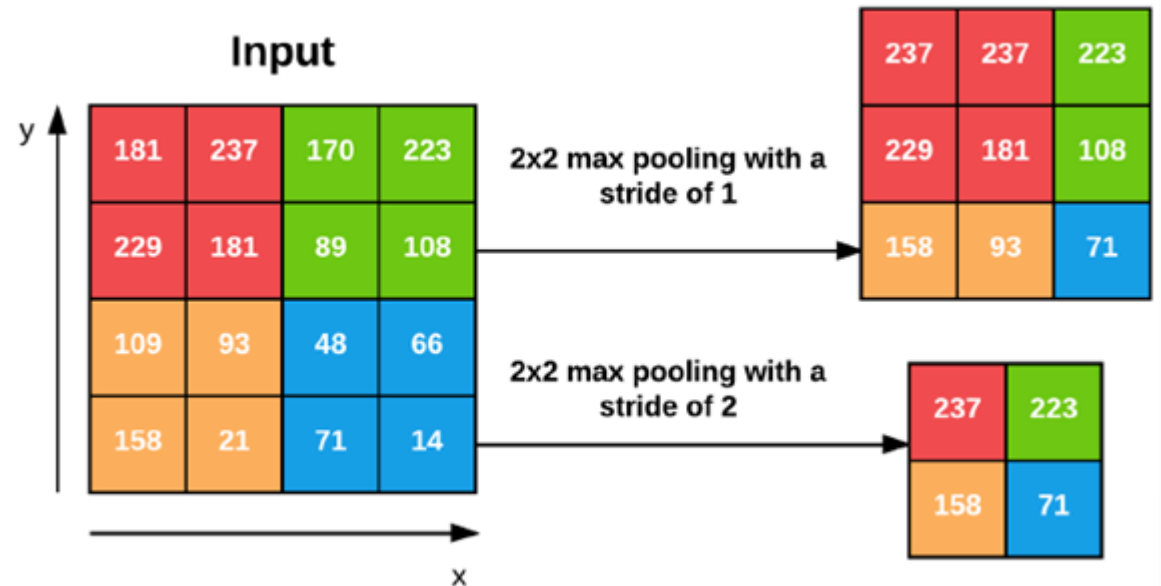
1.정의

3) 풀링(Pooling)

- ① 이미지 분석에서 주로 사용하는 레이어
- ② 이미지에서 특징을 요약하는 역할
 - pool size: 풀링에서 이동하는 윈도우 크기
 - stride: 윈도우가 움직일 때 이동하는 간격

4) GAP(Global average pooling)

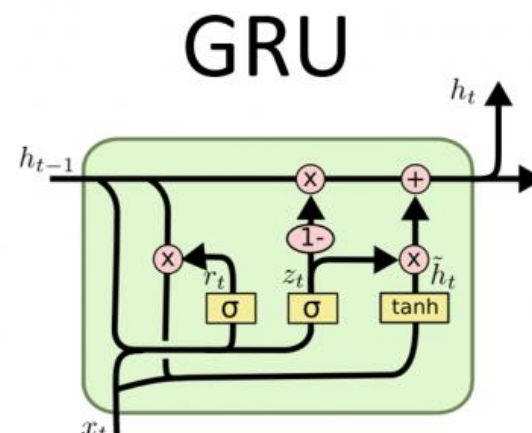
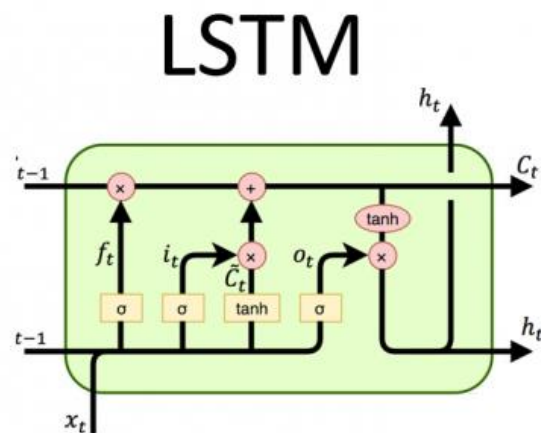
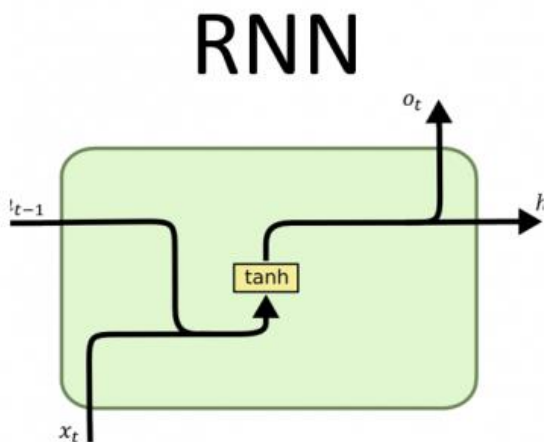
- ① 각 채널별 평균 값을 제시
 - 7x7x1024 크기였으면, 1024를 반환



1.정의

1) RNN(Recurrent neural network)

- ① 순차적인 데이터를 다룰 때 사용하는 레이어
- ② $Y_t = f(Wx_t + UY_{t-1} + b)$ 의 형태(U는 이전 시점의 가중치, f는 활성화함수)
- ③ return_sequences 옵션
 - False: 마지막 시점의 Y_t 제공
 - True: $Y_t(t=0 \sim t)$ 까지 제공

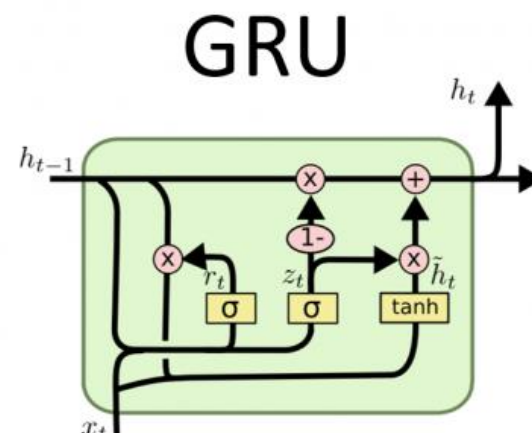
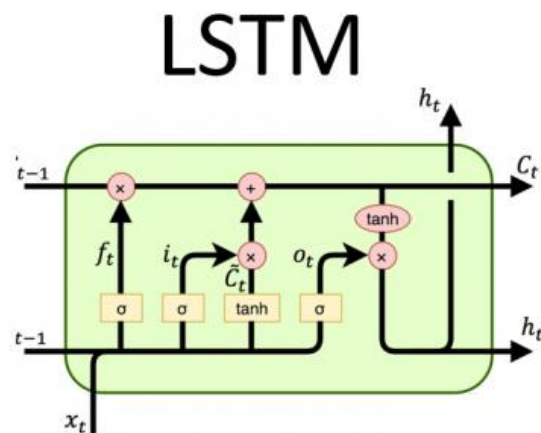
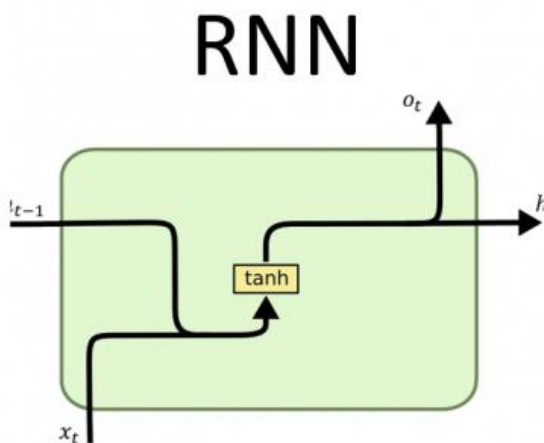


1.정의

2) LSTM(Long short term memory)

① 편의상 k 에 대한 $f(Wx_t + UY_{t-1} + b)$ 를 f_k 라 하겠음

- $c_t = f_f c_{t-1} + f_i f_c$
- $Y_t = f_o \cdot \tanh(c_t)$
- f_i 는 입력 게이트, f_f 는 망각 게이트, f_o 는 출력 게이트, f_c 는 셀 상태 후보를 의미
- f_c 는 현재 시점에서 업데이트 될 정보

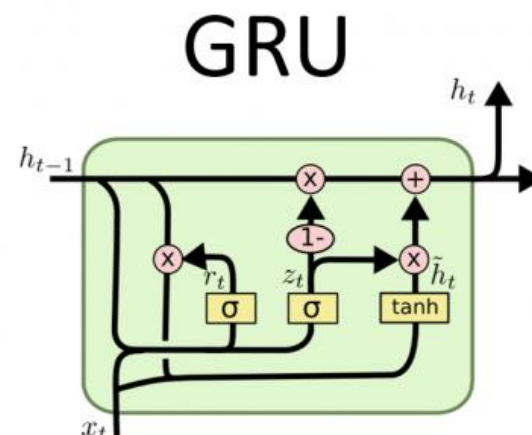
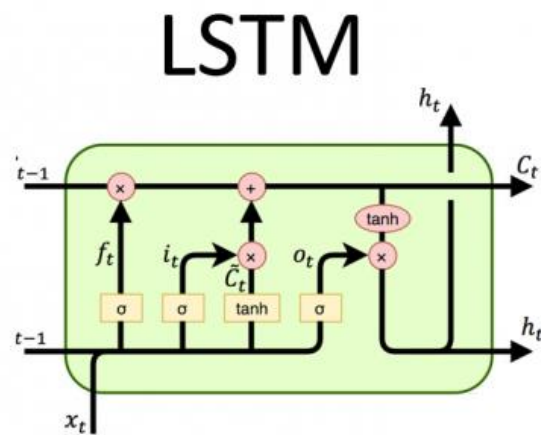
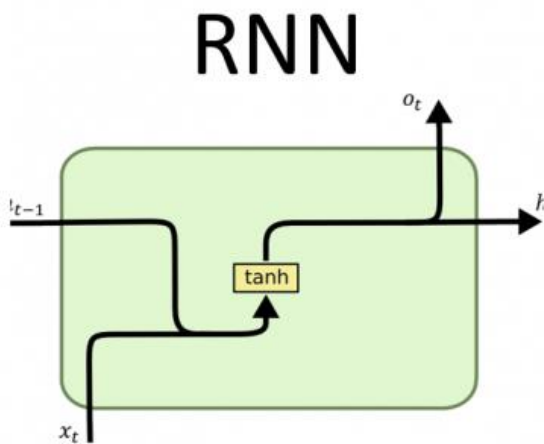


1.정의

3) GRU(Gated recurrent unit)

① 편의상 k 에 대한 $f(Wx_t + UY_{t-1} + b)$ 를 f_k 라 하겠음

- $h_t = \tanh(Wx_t + Uf_r h_{t-1} + b)$
- $Y_t = (1 - f_z)Y_{t-1} + f_z \cdot h_t$
- f_z 는 업데이트 게이트, f_r 은 리셋 게이트, h_t 는 후보 은닉 상태를 의미
- LSTM의 셀 상태를 사용하지 않고 은닉 상태만을 업데이트



1.정의

4) 어텐션(Attention)

- ① 입력데이터에서 중요한 부분에 가중치를 더 주어 모델이 더욱 효과적으로 학습할 수 있도록 하는 기법
- ② 기존 RNN과 LSTM과 같은 순환신경망에서의 장기 의존성 문제가 완화
- ③ 장기 의존성문제: 시퀀스의 길이가 너무 길어지면 초기 데이터에 대한 영향이 소실되는 현상
- ④ 어텐션 과정
 - Query: 현재 처리하고 있는 출력의 일부
 - Key: 입력의 각 부분의 대응값
 - Value: Key의 대응하는 입력부분
 - 가중치 계산
 - 출력값 계산

Query

Key: 검색 결과

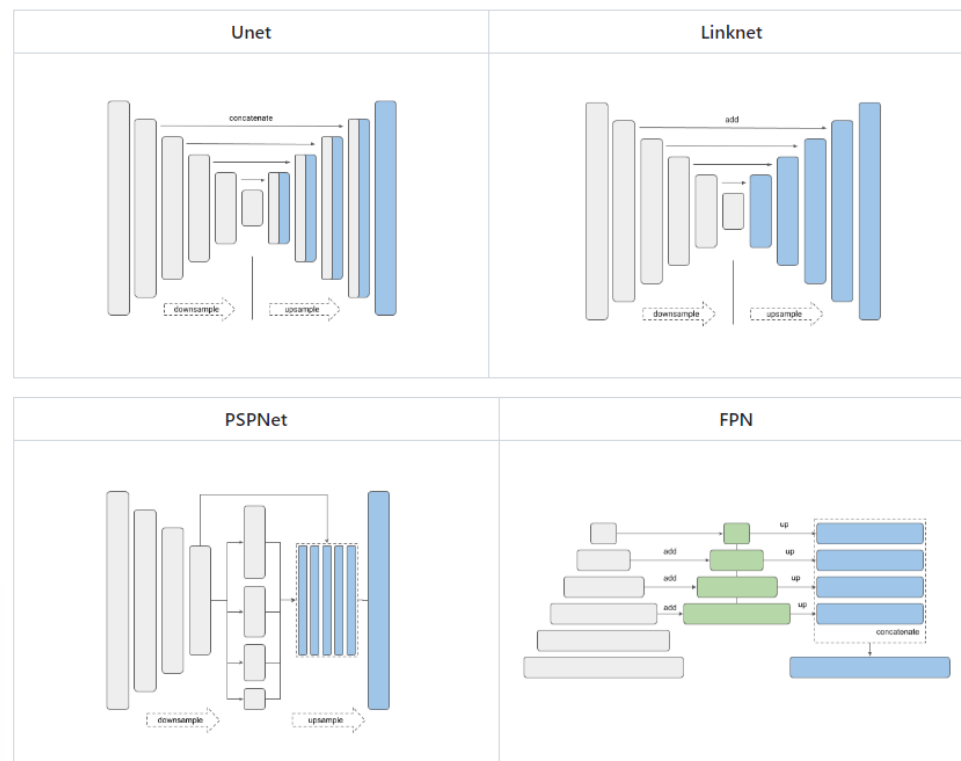
Value: 검색 결과의 관련 내용



1.정의

1) 인코더-디코더(Encoder-Decoder)

- ① 입력 데이터를 압축한 후, 다시 출력 데이터의 형태로 복원하는 형태의 신경망 구조
- ② 중요한 정보를 요약해서 표현하는 벡터를 모델에 따라 다양하게 부름
 - AutoEncoder, GAN: Latent space
 - Unet구조: Bottleneck
 - Seq2Seq: Context Vector
- ③ 인코더-디코더 구조를 다중으로 쌓는 경우도 존재
 - Stacked Hour Glass Network





Thank You

Email: qkdrk777777@naver.com