

Introduction to Convolutional Neural Networks

CNN Basics & Paper Review

20212238 황 병 준

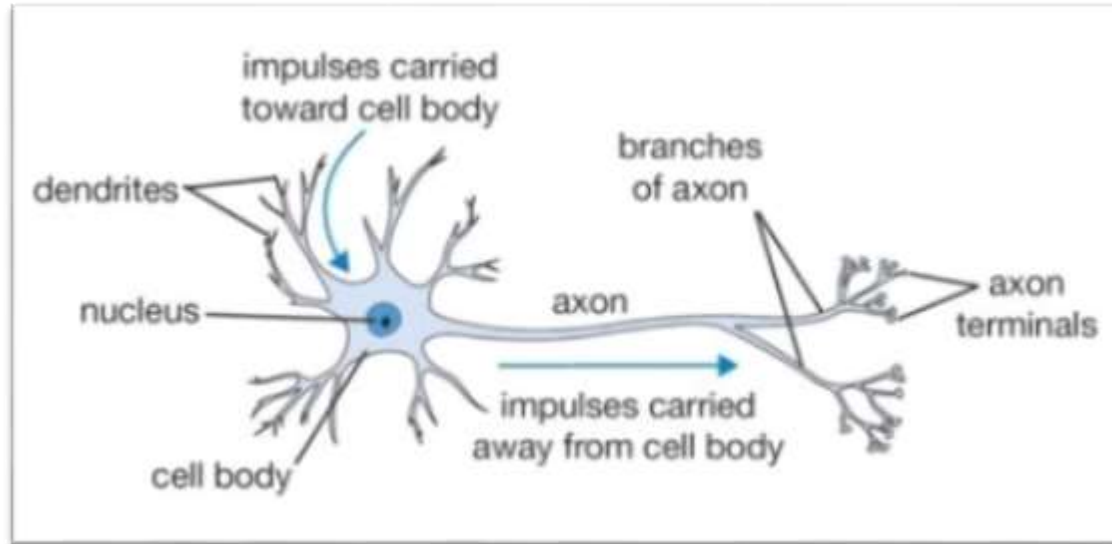
Table of contents

- | | |
|--|---|
| 01 Computer vision algorithm | 07 Key differences of
Between CNN & ANN |
| 02 FNN & RBM, RNN | 08 None |
| 03 Artificial Neural Network | 09 None |
| 04 ANN & CNN | 10 None |
| 05 Overfittig in CNN | 11 None |
| 06 How to Treat Overfitting
in CNN | 12 Receptive Field |
| | 13 Dog & Cat Classification project |

Artificial Neural Network

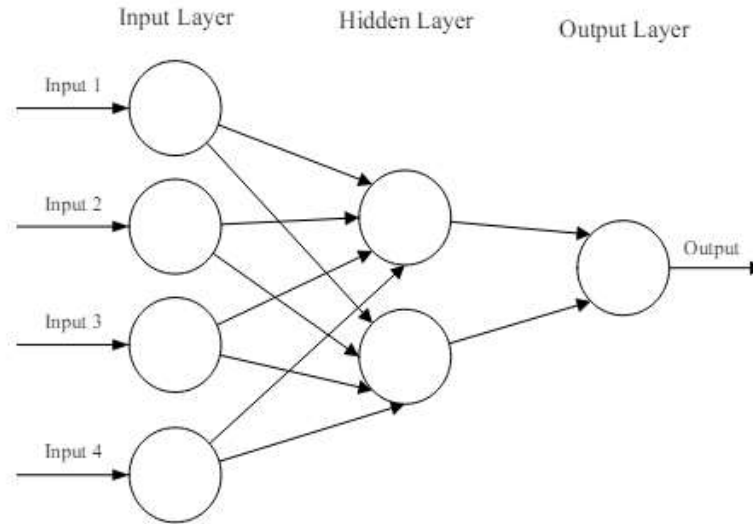
Basic concepts & Operations of ANN

Artificial Neural Network?



ANN & CNN

Basic concepts & Operations of ANN



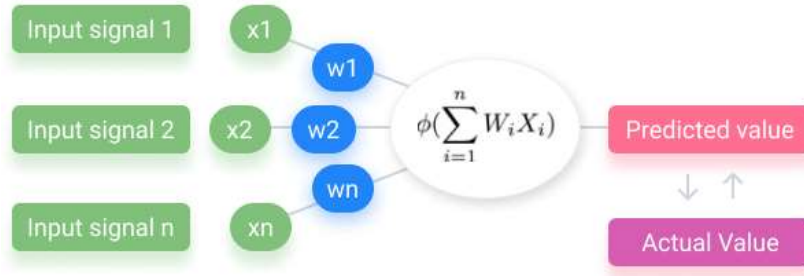
Similarities and Differences of ANN vs. CNN

ANN → Using Weights and activation function

CNN → Using Layer
(Input, Convolutional, Activation, Pooling, Fully-Connected Layer)

Similarities and Differences of ANN vs. CNN

Input Processing Differences



ANN: Feed-Forward Neural Network

CNN: Image data \rightarrow Filter \rightarrow feature map

Similarities and Differences of ANN vs. CNN

ANN

Structural Difference: Fully-Connected Layer 구성

Use of spatial information: image data → Flatten 1D Vector → Spatial information loss

Parameter Sharing: 각 연결에는 가중치가 있어 많은 Parameter 필요

translation invariance: Input data의 평행 이동 취약

Feature extraction method: 수작업으로 직접 특징 추출

Similarities and Differences of ANN vs. CNN

CNN

Structural Difference: Convolutional Layer, Pooling Layer

Use of spatial information: CNN → Convolution Operation → Spatial Info & Regional Patterns 유지

Parameter Sharing: Sharing Convolution Filter → Reduce Number of Parameter

invariance(불변성): Image 위치 변화 강함

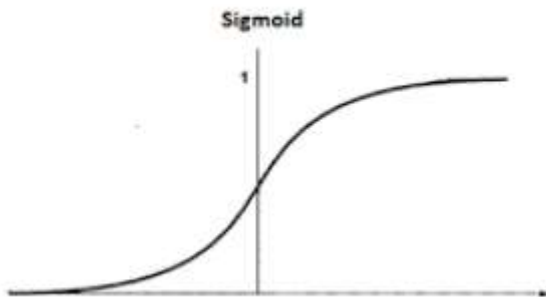
Feature extraction method: Data로 부터 자동으로 특징을 “학습” & “추출”

Convolution Neural Network Important Concepts

Activate Function

Sigmoid

Sigmoid 함수는 input값이 커지면 커질수록 1에 수렴하고, 작아지면 작아질수록 0에 수렴하는 성질을 지녔다.

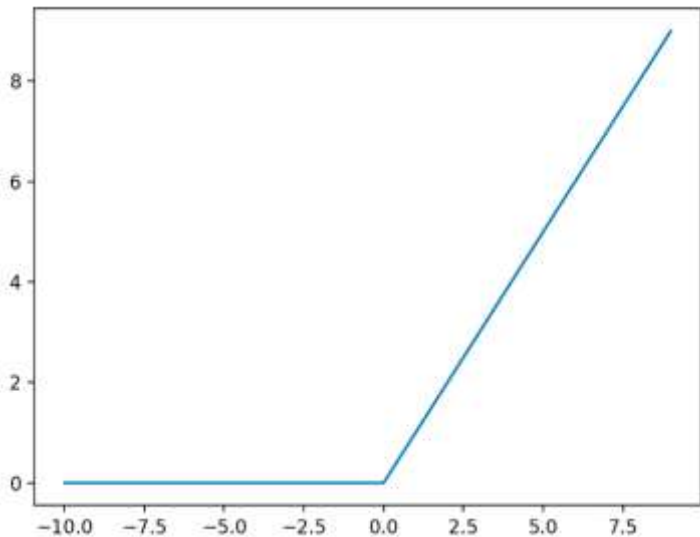


$$\frac{1}{1 + e^{-x}}$$

Activate Function

ReLu

ReLU 함수는 은닉층의 활성화 함수로 사용된다.
입력값이 0보다 작거나 같을 때는 항상 0을 출력, 0보다 크면 입력값과 동일한 출력값 출력



Activate Function

Softmax

Softmax 함수는 시그모이드와 비슷하게, 0 ~ 1사이로 변환하여 출력하지만, 출력값들의 합이 1이 되도록 하는 함수이다.

$$y_k = \frac{\exp(a_k)}{\sum_{i=1}^n \exp(a_i)}$$

정리

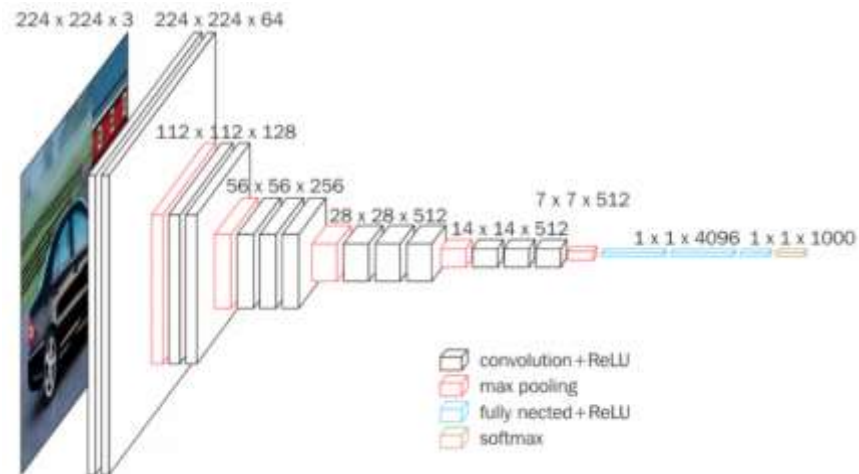
Sigmoid: 이진 분류 모델의 마지막 활성화 함수

Softmax: 다중 분류 모델의 마지막 활성화 함수

ReLU: 기본적으로 은닉층에 사용하는 활성화 함수

Pooling Role in CNN

NONE



Term

Definition and concept of Pooling

Why do we use Pooling?

- First: Reduced Spatial Dimensions
- Second: Feature Translation Invariance
- Third: Preserving information
- Fourth: Control Over Model Size

Term

Reduced Spatial Dimensions

Pooling은 공간적 차원(W, H)을 줄여 신경망에서 계산 비용, 과적합을 줄인다.



Prevent Overfitting

Term

Control Over Model Size



Apply Padding



3 X 3 Kernel, Zero padding

0	0	0	0	0
0				0
0				0
0				0
0	0	0	0	0

5 X 5 출력
생성

Term

Control Over Model Size

즉 Control Over Model Size을 통해서

Feature Map의 Size를 원하는대로 유지 또는 축소 / 확대가 가능하다.



- 제한된 컴퓨팅 자원에서 작은 Model로 학습 가능.
- 특정 작업에서 적합한 모델 크기 선택 가능
- 모델 최적화를 위해 다양한 구조를 탐색 가능

Term

Max Pooling

Max Pooling: Input Feature Map의 각 영역에 대해 최대값을 해당 영역으로 출력으로 사용



각 영역에 존재하는 특징에 대한 가장 중요한 정보를 유지하는데 좋음.

Term

Definition of Pooling

Three Types of Pooling operations

- Max Pooling: 배치의 최대 픽셀 값 선택된다.
- Min Pooling: 배치의 최소 픽셀 값이 선택된다.
- Average Pooling: 배치에 포함된 모든 픽셀의 평균 값이 선택된다.

Term

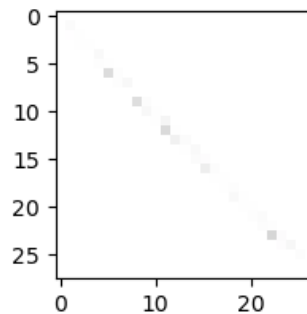
Max pooling

Max Pooling: image의 배경이 어둡고 이미지의 Bright Pixel에 focus 하고싶을 때의 경우 유용함.

Original Image



Max Pooling



Term

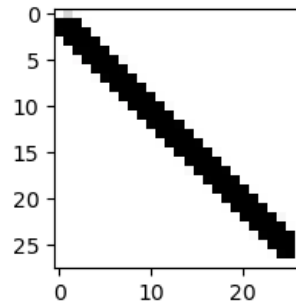
Min pooling

Min Pooling: Max Pooling 과 반대로 이미지의 Dark Pixel에 초점을 맞추고 싶을때 유용함.

Original Image



Min Pooling



Term

Average Pooling

Average Pooling: 각 영역에 있는 요소의 평균값을 계산하여 출력으로 사용



Average Pooling은 때때로 최대 풀링에 비해 더 부드러운 결과 제공 가능함

Term

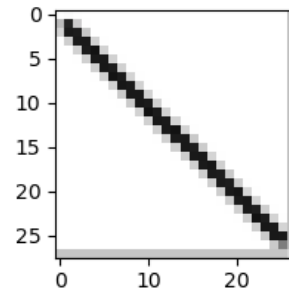
Average pooling

Average Pooling: image를 부드럽게 만들어주기 때문에 선명한 특징이 식별되지 않을수 있음.

Original Image



Average Pooling



Term

Max Pooling

Why use Max Pooling in CNN

계산 감소: 신경망의 계산 복잡성을 제어하여 더 빠르고 효율적으로 수행이 가능하다.

특징 선택: Feature Map에서 Max 값을 선택하기 때문에 Feature이 앞으로 이동하면서 특징 보존 가능

향상된 일반화: 데이터 노이즈 줄임, 최대값만에 focus가 가능함으로 더 중요한 Feature에 집중 가능

과적합 감소: 과적합 방지 가능

Why use Batch Normalization in CNN

Batch Norm: 신경망의 학습 속도 증대, 과정합 방지

- Why is it important?
- How does it work?

Normalization

Normalization: Data를 표준화 하는데 사용되는 전처리 기술

Methods of Normalization

method: 0 ~ 1 사이의 범위로 크기를 조정하는 것

Normalization

How does it Work?

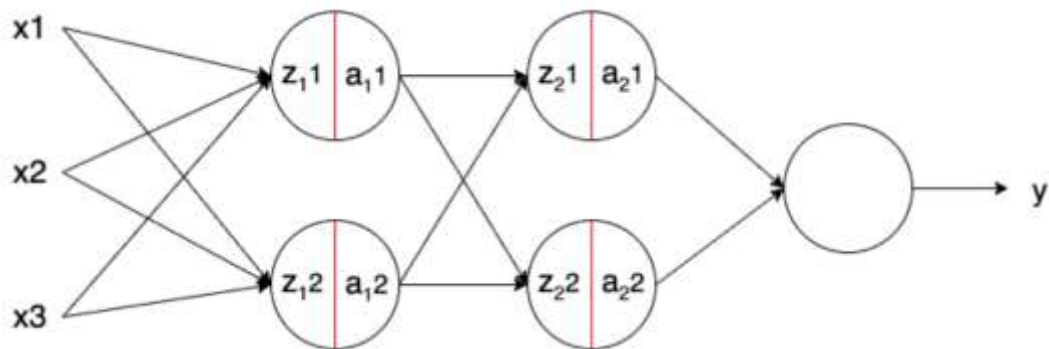
x_i : 입력

z : 뉴런의 출력

a : 활성화 함수의 출력

y : 네트워크 출력

feed forward neural network



Normalization

Batch Norm 적용: 활성화 함수를 적용하기 직전 뉴런의 출력값들에 적용됨

Apply Change State

- 미니배치 내 출력값들의 평균을 뺀다.
- 미니배치 내 출력값들의 표준편차로 나눈다.
- 학습 가능한 스케일링 파라미터(γ)를 곱한다.
- 학습 가능한 이동 파라미터(β)를 더한다.

Result: 평균, 표준편차 0, 1로 변환됨.

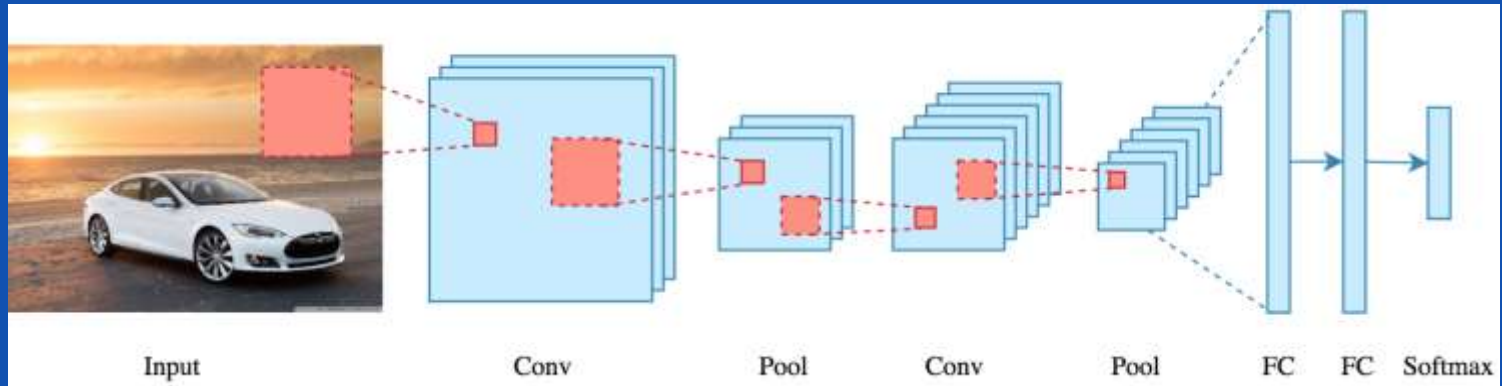
CNN Architecture

Detailed operations

CNN Architecture

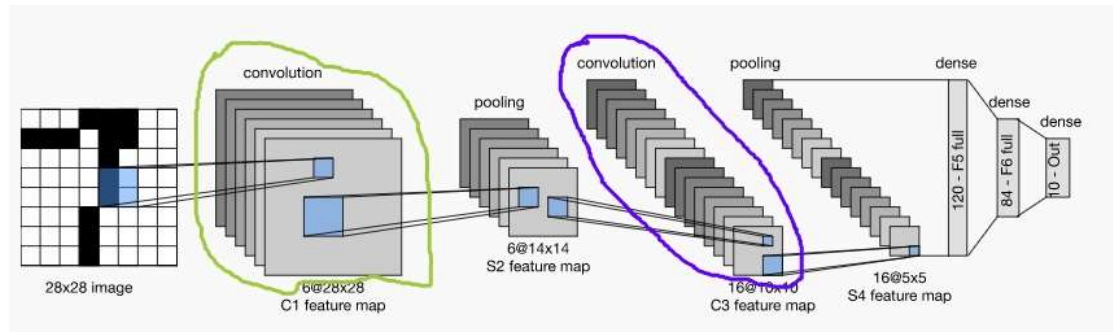
Overall Architecture

Simple CNN architecture Consisting of 5 Layers



CNN Architecture

Convolution layer에는 많은 filter 존재 → 특징 추출



CNN Architecture

First Convolution Layer

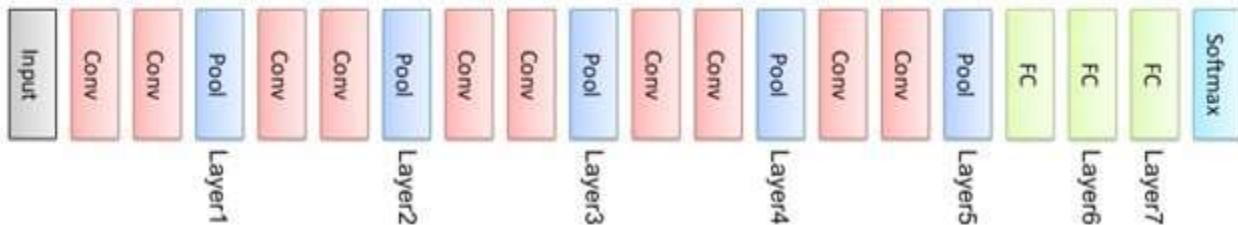
Edge filter: CNN에서의 첫번째 작업



CNN Architecture

Pooling layer

Conv Layer 사이 Pooling Layer는 어떤 역할을 하나요?



CNN Architecture

0	0	0	0	0	0
0	105	102	100	97	96
0	103	99	103	101	102
0	101	98	104	102	100
0	99	101	106	104	99
0	104	104	104	100	98

Image Matrix

Kernel Matrix		
0	-1	0
-1	5	-1
0	-1	0

320				

Output Matrix

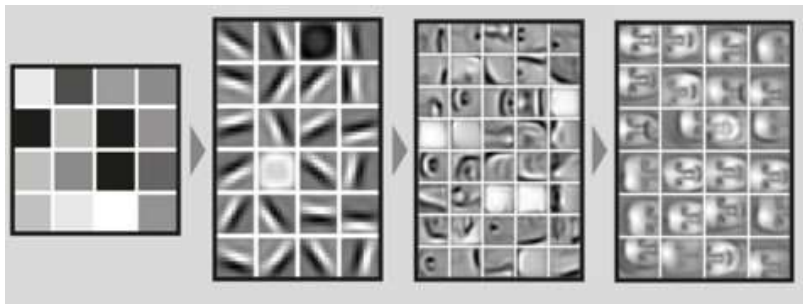
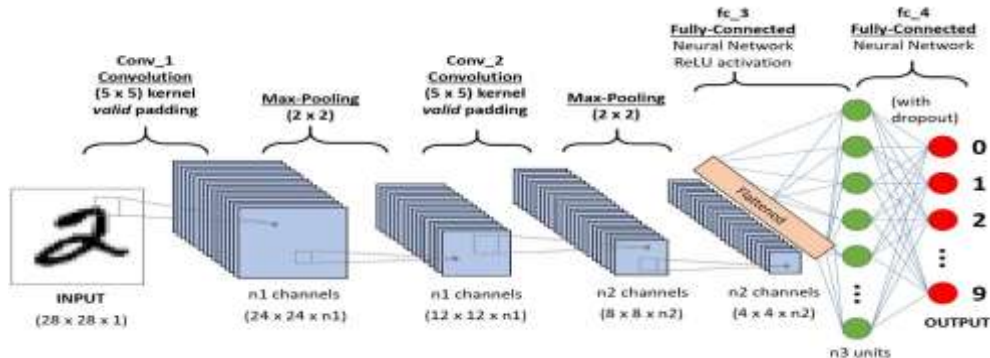
$$\begin{aligned} &0 * 0 + 0 * -1 + 0 * 0 \\ &+ 0 * -1 + 105 * 5 + 102 * -1 \\ &+ 0 * 0 + 103 * -1 + 99 * 0 = 320 \end{aligned}$$

**Convolution with horizontal and
vertical strides = 1**

CNN Architecture

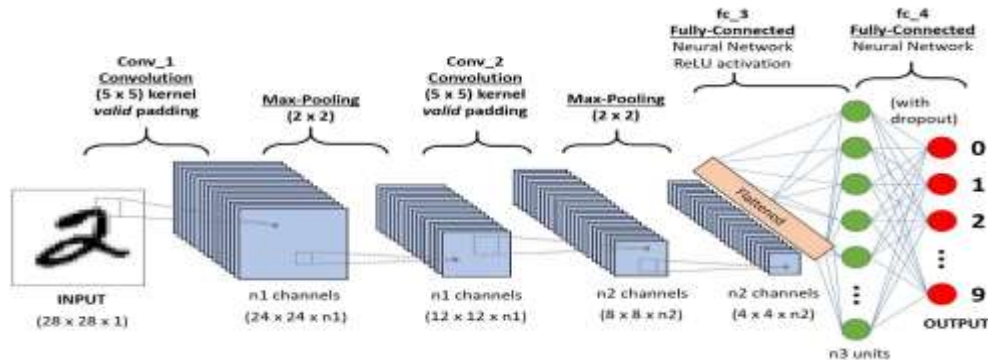
FC layer

Feature를 뽑다보면 마지막에는 물체와 유사한 형태들의 feature map 선별된다.



CNN Architecture

Layer가 뒤로 갈수록 filter의 총 개수의 증가 이유

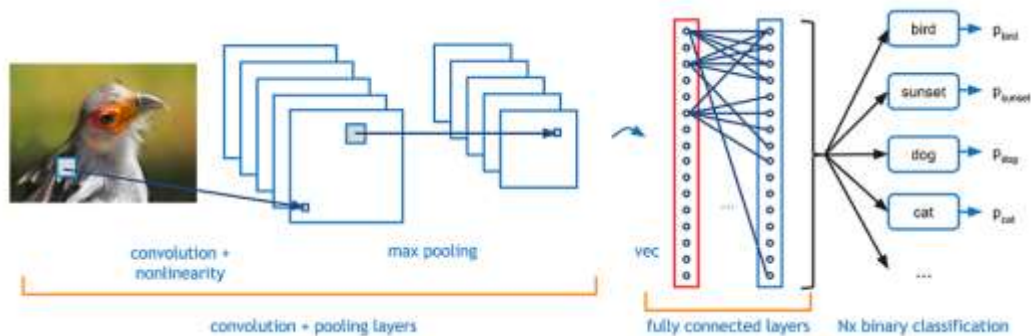


1. 계층적 특징 표현 학습 초기 → 레이어를 넘어갈수록 복잡하고 추상적인 고수준 특징 학습하기 때문
2. 특징 맵 유지 Pooling layer를 거치면서 중요한 패턴을 유지하기 때문 → 정보 손실 방지
3. 정보 용량 증가
4. 과적합 방지

CNN Architecture

Classification

객체와 유사한 Feature map을 통해서 Classification을 하게 됨.



Dog VS Cat Classification project using Convolutional Neural Network





Thanks!

CREDITS: This presentation template was created by **Slidesgo**, and includes icons by **Flaticon** and infographics & images by **Freepik**