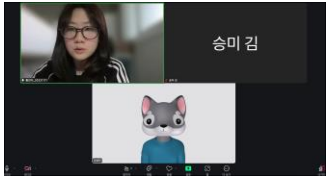
[CUAI] CS231n 스터디 1팀

2025.05.12

발표자 : 김성민

스터디원 소개 및 만남 인증



Study Info

- **스터디 시작일** : 2025.03.17(월) ~ 6/2(월)
- **스터디 장소** : 중앙대학교 스터디룸
- **스터디 시간** : 매주 월요일 1회 18:00~20:00
- **스터디 구성원** : 황민아, 조한서, 김승미, 김성민

목차

1. 스터디 진행 상황
2. 학습 내용

스터디 진행 상황

[1주차] 3/17
1. Introduction to CNN for Visual Recognition
2. Image Classification
3. Loss Functions and Optimization

[2주차] 3/24
4. Introduction to NN
5. CNN
6. Training Neural Networks, Part 1

[3주차] 3/31
7. Training Neural Networks, Part 2
8. Deep Learning Software

중간고사 4/22~28

[4주차] 5/5
9. CNN Architectures
10. RNN

[5주차] 5/12
12. Visualizing and Understanding

[6주차] 5/19
11. Detection and Segmentation
13. Generative Models

[7주차] 5/26
14. Reinforcement Learning
15. Efficient Methods and Hardware for Deep Learning

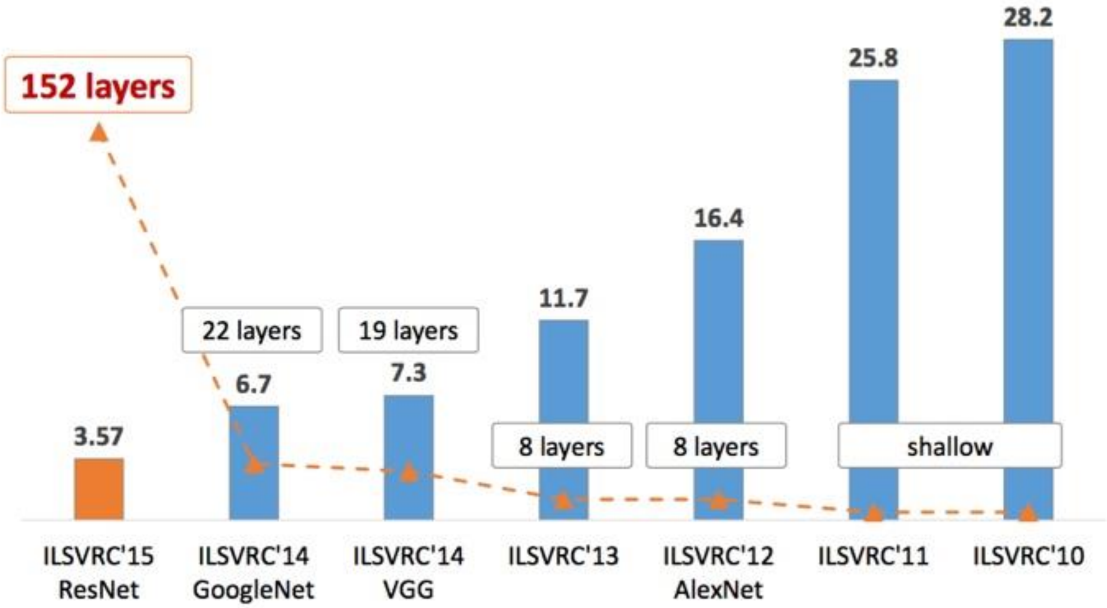
[8주차] 6/2
16. Adversarial Examples and Adversarial Training

기말고사 6/17~23

1) CNN Architecture

- AlexNet
- VGG
- GoogleNet
- ResNet

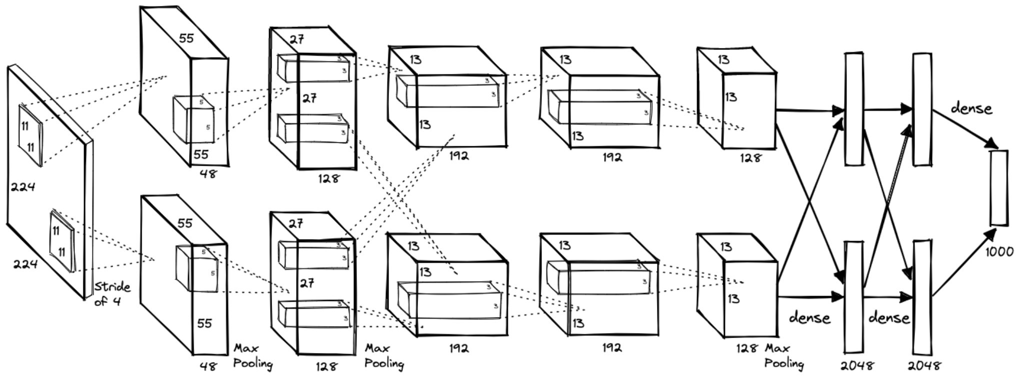
...



1) CNN Architecture

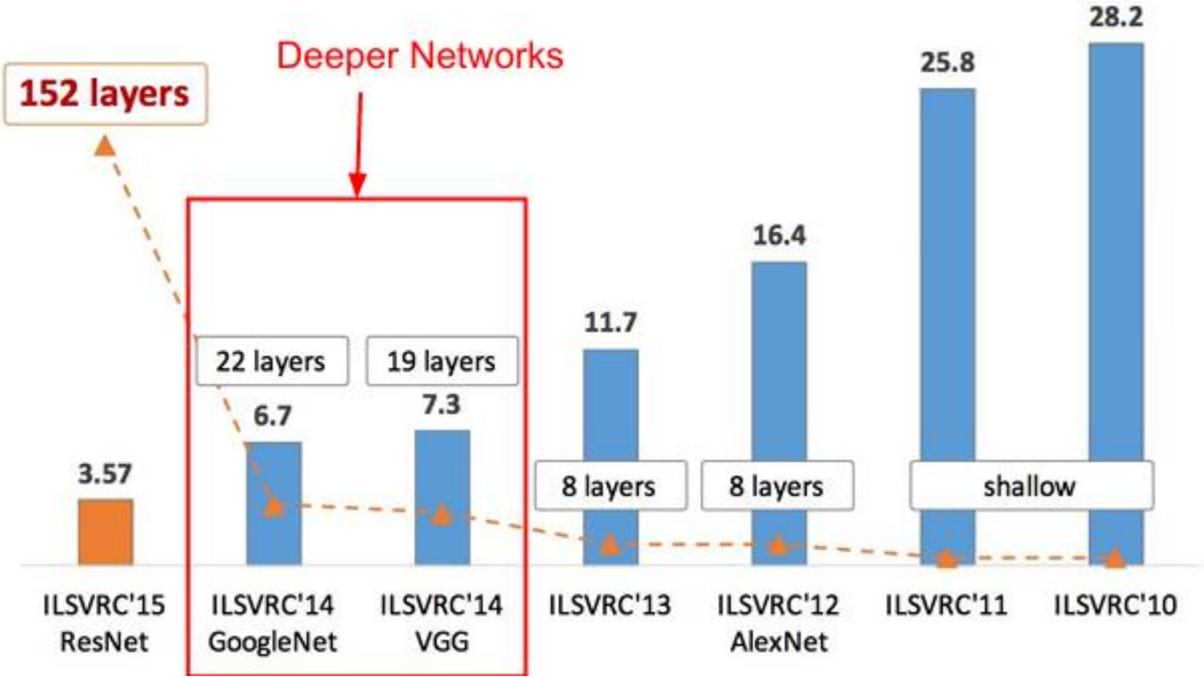
AlexNet

CONV1
MAX
POOL1
NORM1
CONV2
MAX
POOL2
NORM2
CONV3
CONV4
CONV5
Max
POOL3
FC6 FC7
FC8



1) CNN Architecture

VGGNet

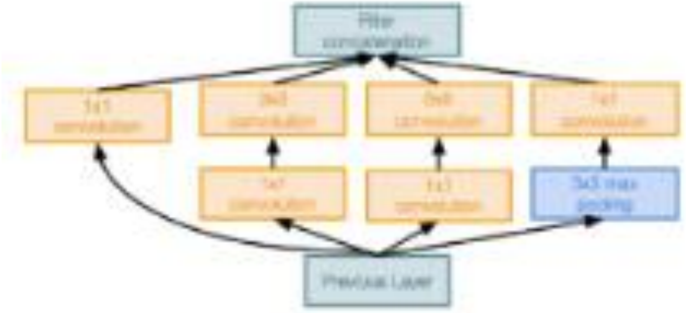


1) CNN Architecture

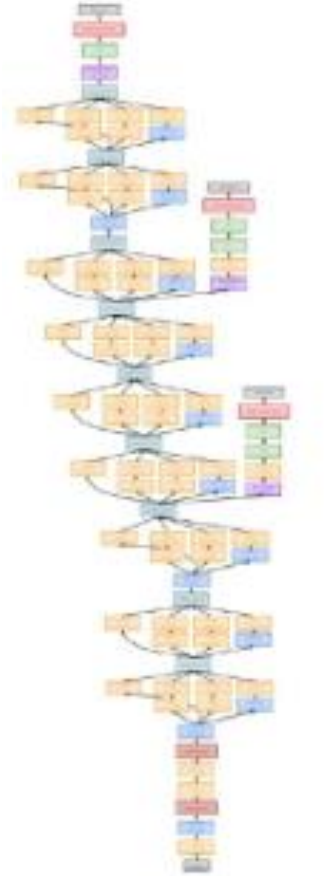
VGGNet

1) CNN Architecture

GoogLeNet

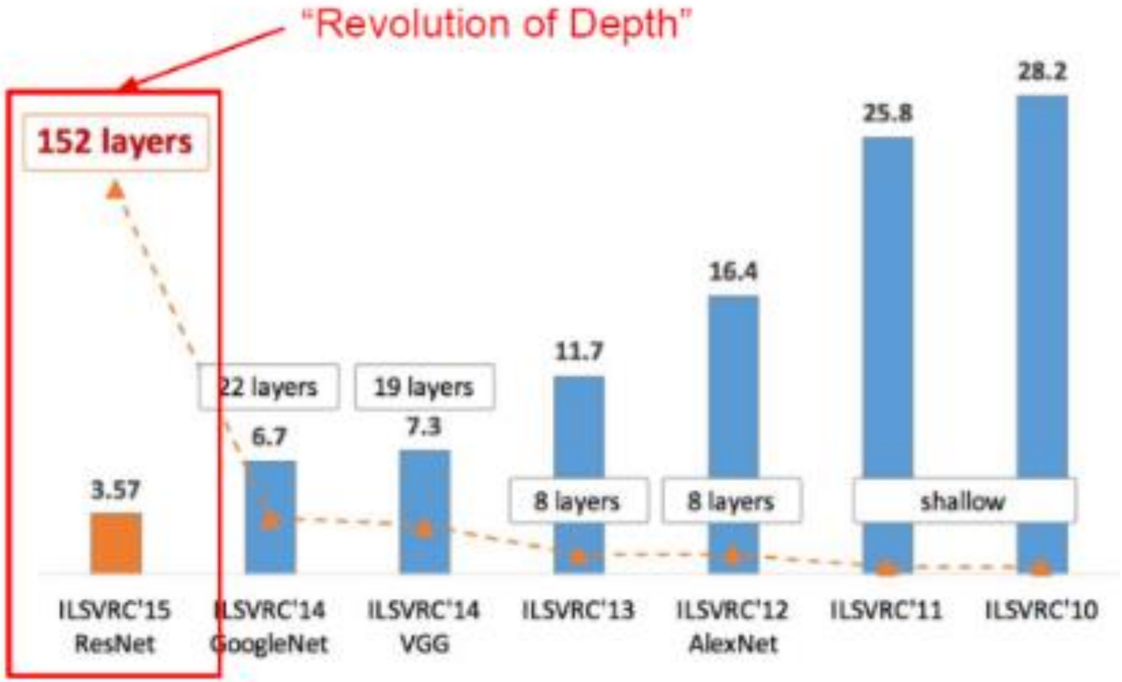


Inception module



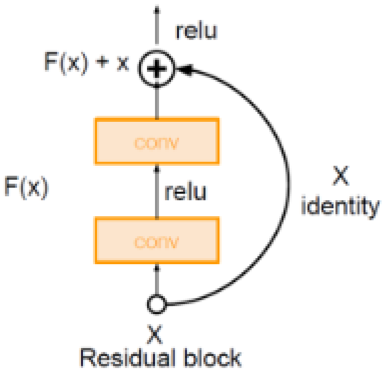
1) CNN Architecture

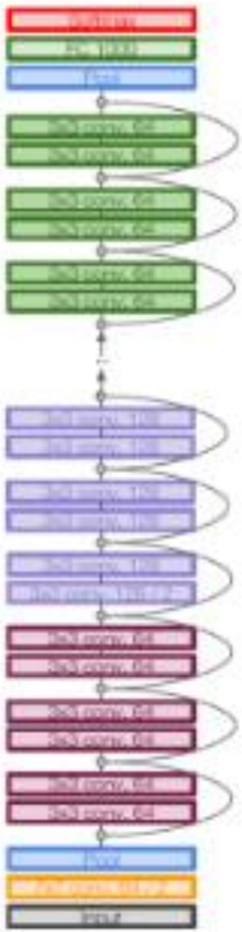
ResNet



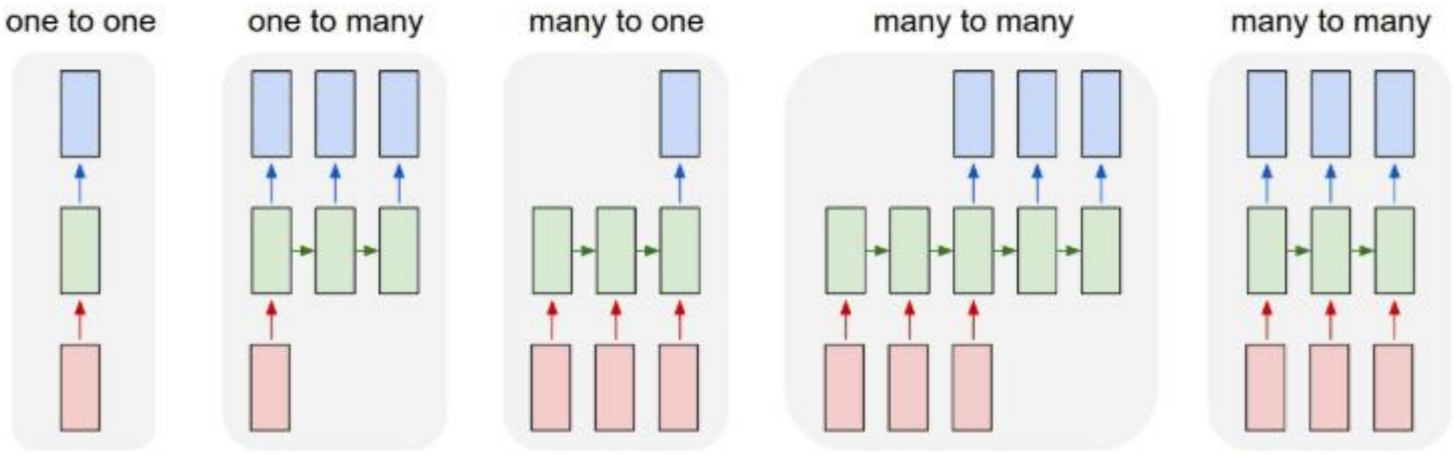
1) CNN Architecture

ResNet

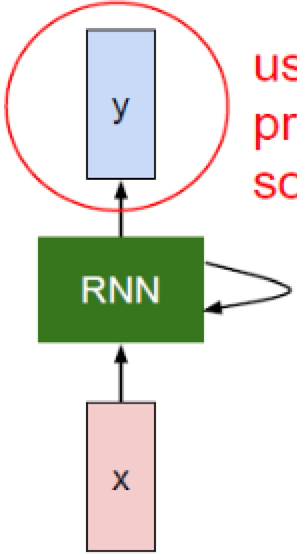




2) Recurrent Neural Network



2) Recurrent Neural Network



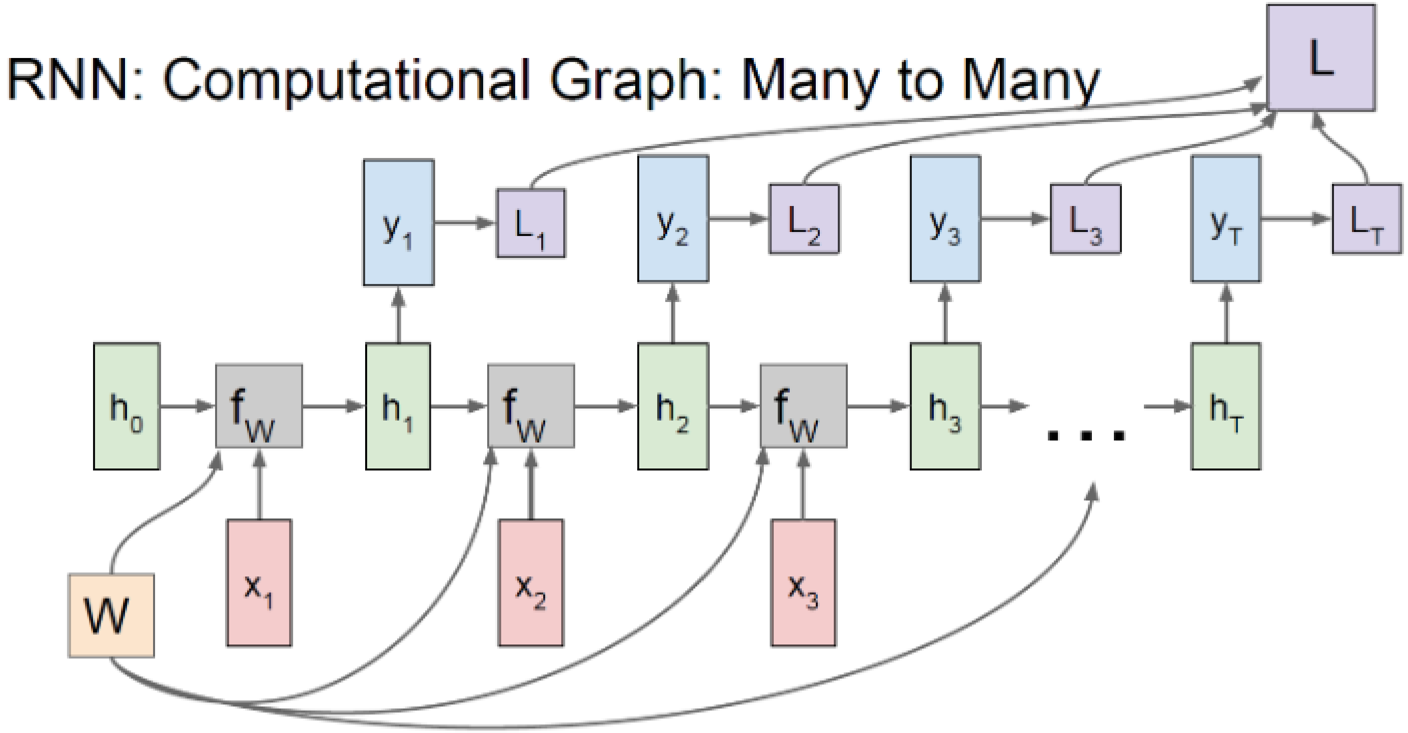
usually want to predict a vector at some time steps

$$\underbrace{\boxed{h_t}}_{\text{new state}} = \underbrace{\boxed{f_W}}_{\text{some function with parameters W}}(\underbrace{\boxed{h_{t-1}}}_{\text{old state}}, \underbrace{\boxed{x_t}}_{\text{input vector at some time step}})$$

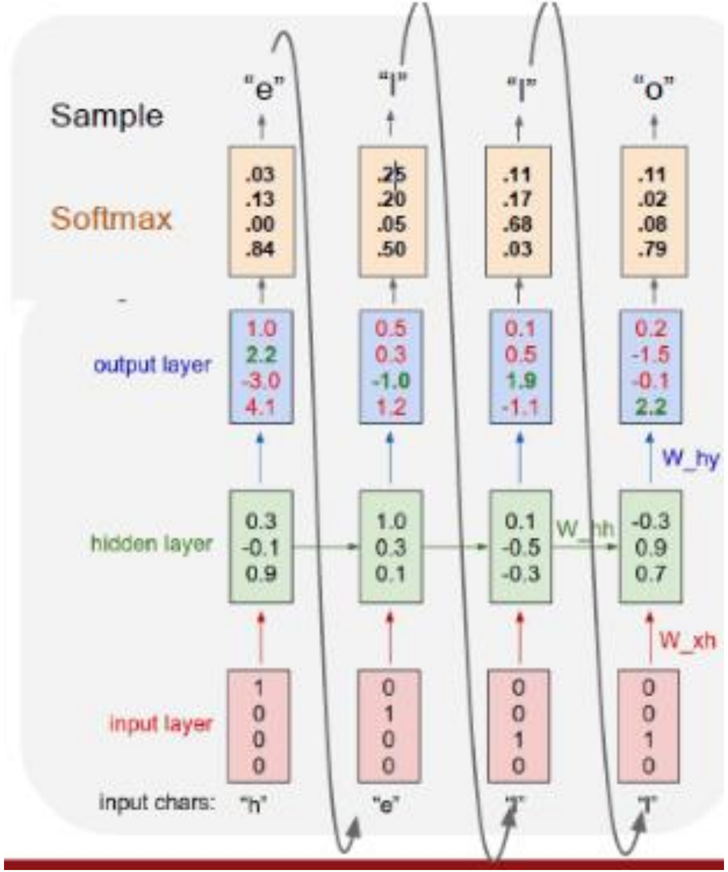
$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

$$y_t = W_{hy}h_t$$

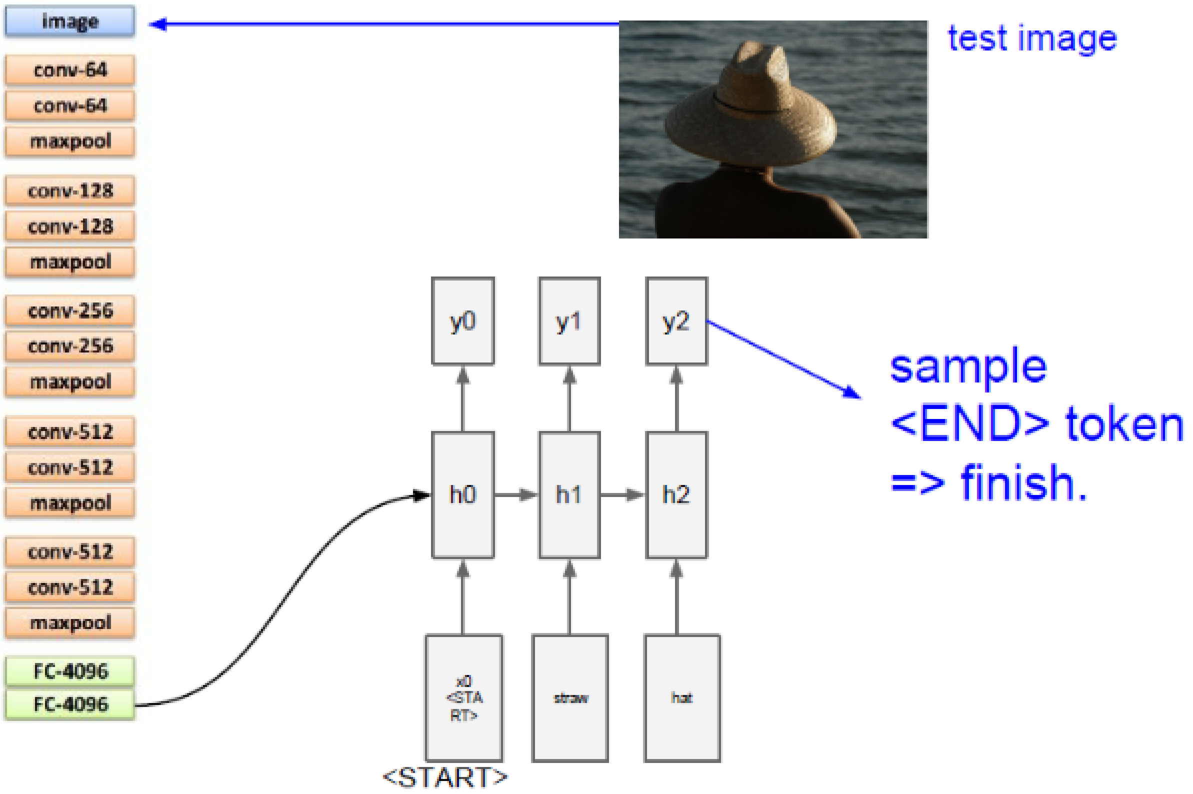
2) Recurrent Neural Network



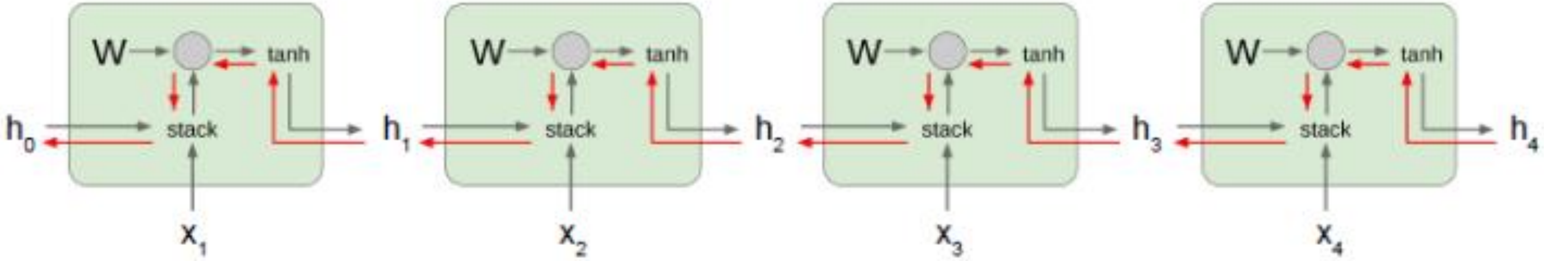
2) Recurrent Neural Network



2) Recurrent Neural Network



2) Recurrent Neural Network

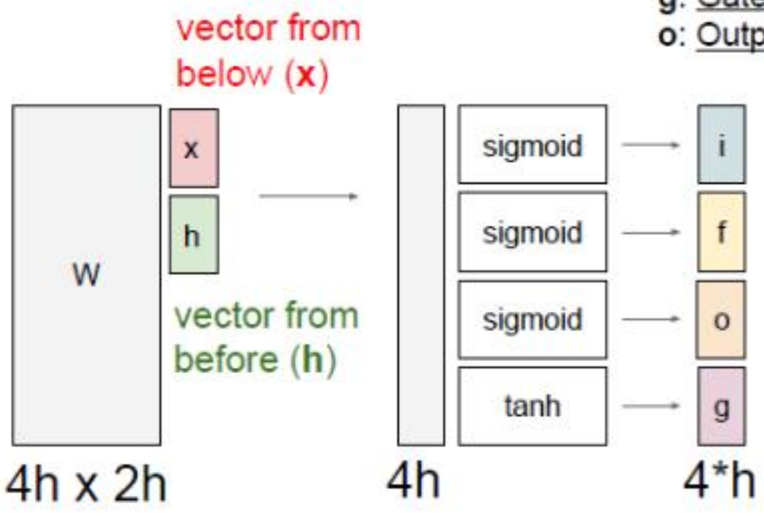


2) Recurrent Neural Network

Long Short Term Memory (LSTM)

[Hochreiter et al., 1997]

f: Forget gate, Whether to erase cell
i: Input gate, whether to write to cell
g: Gate gate (?), How much to write to cell
o: Output gate, How much to reveal cell

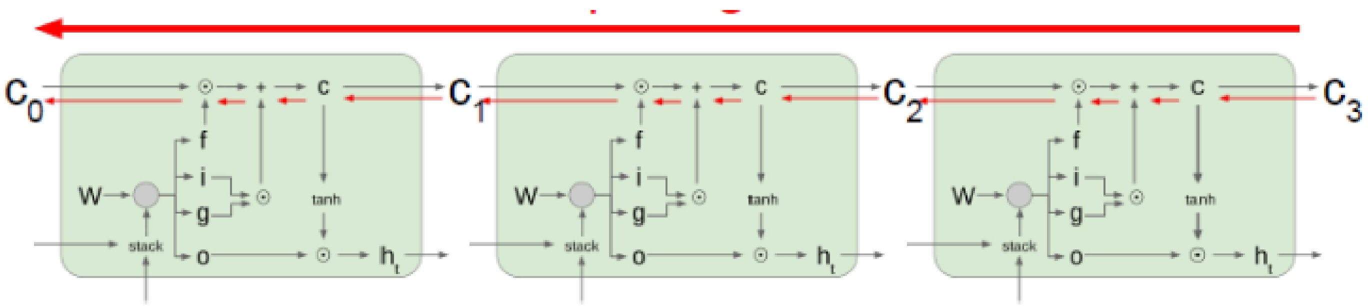


$$\boxed{\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}}$$

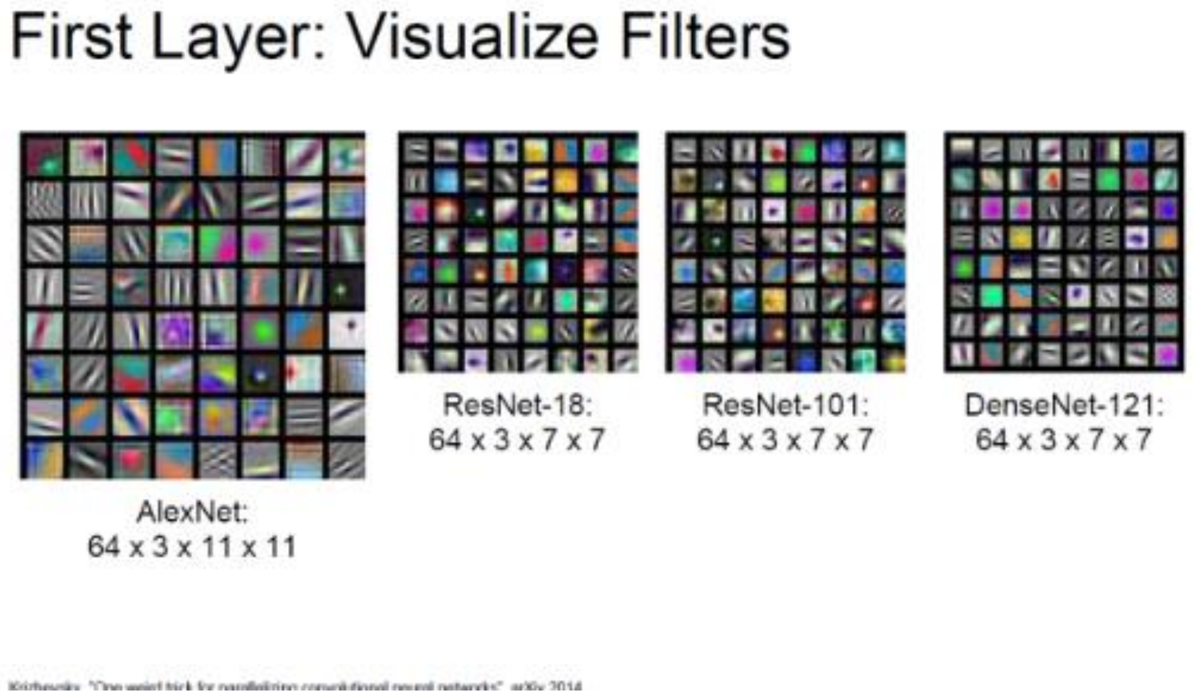
$$c_t = f \odot c_{t-1} + i \odot g$$

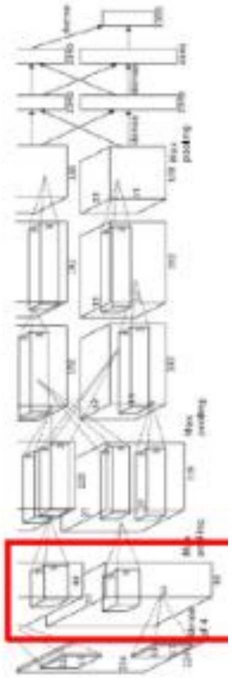
$$h_t = o \odot \tanh(c_t)$$

2) Recurrent Neural Network



3) Visualizing and Understanding





3) Visualizing and Understanding

layer 1 weights

16 x 3 x 7 x 7

layer 2 weights

20 x 16 x 7 x 7

layer 3 weights

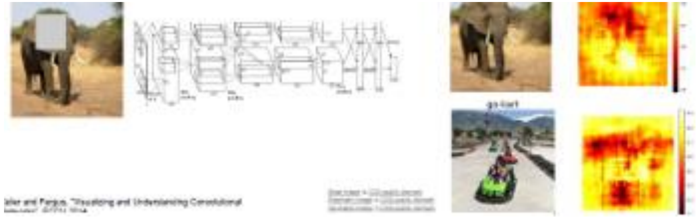
20 x 20 x 7 x 7

3) Visualizing and Understanding

3) Visualizing and Understanding

Maximally Activating Patches

Occlusion Experiments



Saliency Maps

3) Visualizing and Understanding

Gradient Descent

입력 이미지가 들어왔을 때 Weight 값을 update하기 위해 사용했던 방법

Gradient Ascent

해당 뉴런을 활성화시키는 General한 입력 이미지를 찾아내는 방법

$$I^* = \arg\max_I \underbrace{\boxed{f(I)}}_{\text{Neuron value}} + \underbrace{\boxed{R(I)}}_{\text{Natural image regularizer}}$$

감사합니다