CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 1

Read

Solution: Complete ■

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 2

Read

Solution: Complete ■

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 3

Read

Solution: Complete ■

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 4

Read

Solution: Complete ■

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 5

Compute the receptive field size at the input to the global average pooling layer for ResNet 50.

Solution: pass ■

Stride then add

Proof:

$$\left(\left(\underbrace{1+2*3}_{\text{conv5}}+\underbrace{2*6}_{\text{conv4}}+\underbrace{2*4}_{\text{conv4}}+\underbrace{2*3}_{\text{conv4}}\right)\underbrace{*2+2}_{\text{max pool}}\right)\underbrace{*2+6}_{\text{conv1}} \tag{1}$$

$$= 142 \tag{2}$$

□

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 6

(a) How does the accuracy of the half wide version copare to the original version?

Solution: The accuracy at the final epoch for the full and half versions are

$$\text{Full} = 91.03\% \qquad \text{Half} = 89.31\% \tag{3}$$

■

Proof: □

(b) How long does an epoch of training take for both versions?

Solution: The times for epoch 3 for both models (trained on GTX970) are

$$\text{Full} = 59s \qquad \text{Half} = 36s \tag{4}$$

■

Proof: □

(c) Approximate how feature map memory, filter memory, and compute change between the full and half width versions.

Solution:

$$\text{Feature map memory} = \frac{1}{2} \tag{5}$$

$$\text{Filter memory} = \frac{1}{2} \tag{6}$$

$$\text{Compute} = \frac{1}{4} \tag{7}$$

■

Proof: As an approximation we can consider how CNN style 2D convolution layers will be impacted by the width reduction, as Resnet is primarily composed of such layers. For filter and feature map memory we will see a reduction approximately equal to the reduction in width. The dimensions of the filters and feature maps remain unchanged, but we are only considering half of the original feature maps and filters.

Recall that compute for CNN style 2D convolution depends on $N_o * N_i$ which are both reduced by half, leading to a total reduction of $1/4$. □

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 7

Solution: The network design will be based on Inception v4. To inform our choice of network architecture we can use the given hint for an expected input global average pooling and think backwards from here. Obviously for CIFAR-10 we will have 10 classes, so we need to choose N_i to global average pooling somewhere above 10, call it 50.

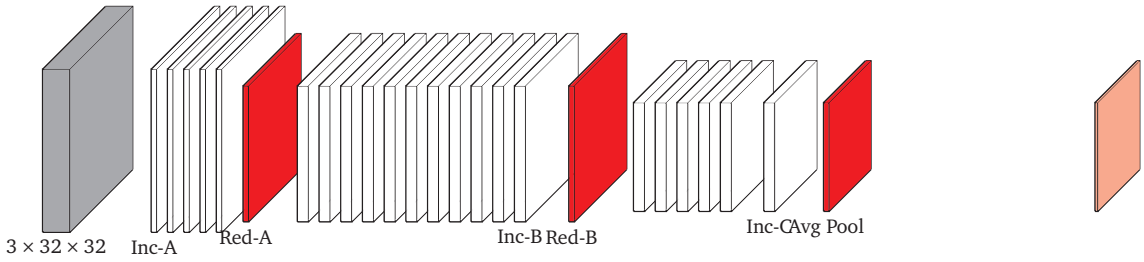

Figure 1. Example CNN.

■

Proof: □

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 8

Solution: ■

Proof: □

CS 6301.503 Spring 2019 Scott C. Waggener (scw180000)

Homework 6 (Design) Problem 9

Solution: ■

Proof: □