CSE 151B Deep Learning
UC San Diego, Spring 2022

Homework 2

Name: William Sun UID: A16013590

1 Convolutional Neural Networks

Problem A: Convolution

i.

Solution A.i:

8 filters, each with $5 \times 5 \times 3$ *weights plus a bias term per filter is equal to* $(8 \times 5 \times 5 \times 3) + 8 = 608$ *parameters.*

ii.

Solution A.ii:

$W_{out} = H_{out} = (W - F + 2P)/S + 1 = (32 - 5 + 0)/1 + 1 = 28$. *Thus, the output tensor has shape* $(28, 28, 8)$ *where* 8 *is based on the number of filters.*

Problem B: Pooling

i.

Solution B.i:

Calculating average using floating point numbers:

$$\begin{bmatrix} 1 & 0.5 \\ 0.5 & 0.25 \end{bmatrix}$$

$$\begin{bmatrix} 0.5 & 1 \\ 0.25 & 0.5 \end{bmatrix}$$

$$\begin{bmatrix} 0.25 & 0.5 \\ 0.5 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0.5 & 0.25 \\ 1 & 0.5 \end{bmatrix}$$

ii.

Solution B.ii:

All 4 matrices have the same result:

$$\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

2 Recurrent Neural Networks

Problem A: LSTM

i.

Solution A.i:

Use cross-entropy loss: $L_t = -y_t \log \hat{y}_t$

where $\hat{y}_t = \sigma(w_y h_t)$

1) $$\delta c_t = \frac{\partial L_t}{\partial c_t} = \underbrace{\frac{\partial L_t}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial h_t}}_{(\hat{y}_t - y_t) w_y^T} \cdot \frac{\partial h_t}{\partial c_t}$$

$$\frac{\partial (o_t \odot \tanh(c_t))}{\partial c_t} = o_t \odot \tanh'(c_t) = o_t (1 - \tanh^2(c_t))$$

Thus, $\boxed{\delta c_t = (\hat{y}_t - y_t) w_y^T \cdot o_t (1 - \tanh^2(c_t))}$

2) $$\delta o_t = \frac{\partial L_t}{\partial o_t} = \underbrace{\frac{\partial L_t}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial h_t}}_{(\hat{y}_t - y_t) w_y^T} \cdot \underbrace{\frac{\partial h_t}{\partial o_t}}_{\tanh(c_t)}$$

Thus, $\boxed{\delta o_t = (\hat{y}_t - y_t) w_y^T \cdot \tanh(c_t)}$

3) $$\delta i_t = \frac{\partial L_t}{\partial i_t} = \underbrace{\frac{\partial L_t}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial h_t} \cdot \frac{\partial h_t}{\partial c_t}}_{\text{from (1)}} \cdot \underbrace{\frac{\partial c_t}{\partial i_t}}_{\tanh(w^{cx} x_t + w^{ch} h_{t-1})}$$

Thus, $\boxed{\delta i_t = (\hat{y}_t - y_t) w_y^T \cdot o_t (1 - \tanh^2(c_t)) \cdot \tanh(w^{cx} x_t + w^{ch} h_{t-1})}$

4) $$\delta f_t = \frac{\partial L_t}{\partial f_t} = \underbrace{\frac{\partial L_t}{\partial \hat{y}_t} \cdot \frac{\partial \hat{y}_t}{\partial h_t} \cdot \frac{\partial h_t}{\partial c_t}}_{\text{from (1)}} \cdot \underbrace{\frac{\partial c_t}{\partial f_t}}_{c_{t-1}}$$

Thus, $\boxed{\delta f_t = (\hat{y}_t - y_t) w_y^T \cdot o_t (1 - \tanh^2(c_t)) \cdot c_{t-1}}$

ii.

Solution A.ii:

Based on the derivations above, the gradient does not explode if the forget gates are close to 1 and the input/output gates are close to 0. Looking at the gradient of c_t*, if the output gate is close to zero, the gradient approaches 0 too, which means the system remains stable and unchanged. Additionally, since the forget gate is close to 1 and input is close to 0, the cell state will keep its previous value, thereby preventing both an exploding gradient and vanishing gradient.*

3 Poem Generation

Problem A: Pre-processing

i.

Solution A.i: *Your solution here*

ii.

Solution A.ii: *Your solution here*

Problem B: Model Training

i.

Solution B.i: *Your solution here*

ii.

Solution B.ii: *Your solution here*

iii.

Solution B.iii: *Your solution here*