Übungsblatt: 4

Bearbeitung am 14. Juni

Aufgabe 1: Fully Connected

Gegeben ist eine Gewichtsmatrix w, ein Biasvector b und eine Eingabe x einer Fully Connected Layer, wie in der Vorlesung besprochen:

$$w = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix}$$

$$b = \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$$

$$x = \begin{bmatrix} 1 & 2 \end{bmatrix}$$

(a) Berechnen Sie den Forward-Pass

Lösung:

$$y = x \cdot w + b$$

$$y = \begin{bmatrix} 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix} + \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$$

$$y = \begin{bmatrix} 5 & 11 & 17 \end{bmatrix} + \begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$$

$$y = \begin{bmatrix} 6 & 13 & 20 \end{bmatrix}$$

(b) Berechnen Sie den Vektor δx. Gegeben ist folgender Fehlervektor $\delta y = \begin{bmatrix} 0 & 0.5 & 1 \end{bmatrix}$

Lösung:

$$\delta x = \delta y \cdot w^T$$

$$w^T = \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

$$\delta x = \begin{bmatrix} 0 & 0.5 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix}$$

$$\delta x = \begin{bmatrix} 6.5 & 8 \end{bmatrix}$$

(c) Berechnen Sie die Gewichtsupdates ΔW und Δb für die obigen Angaben.

Lösung:

$$\Delta w = x^T \cdot \delta y$$
$$\Delta w = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0.5 & 1 \end{bmatrix}$$
$$\Delta w = \begin{bmatrix} 0 & 0.5 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$
$$\Delta b = \delta y$$

Aufgabe 2: Convolution-2D

Es sei die Convolution-2D Layer wie in der Vorlesung eingeführt.

(a) Verändern Sie den Filter f (mit 2 Channels)

$$f_0 = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix} \begin{bmatrix} 7 & 9 & 11 \\ 8 & 10 & 12 \end{bmatrix} f_1 = \begin{bmatrix} 13 & 15 & 17 \\ 24 & 16 & 18 \end{bmatrix} \begin{bmatrix} 19 & 21 & 23 \\ 20 & 22 & 24 \end{bmatrix}$$

, so dass er für den Backward-Pass genutzt werden kann.

Lösung: Vertausche Channels und Filter:

$$f_0 = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix} \begin{bmatrix} 13 & 15 & 17 \\ 24 & 16 & 18 \end{bmatrix} f_1 = \begin{bmatrix} 7 & 9 & 11 \\ 8 & 10 & 12 \end{bmatrix} \begin{bmatrix} 19 & 21 & 23 \\ 20 & 22 & 24 \end{bmatrix}$$

Dann um 180 Grad rotieren:

$$f_0 = \begin{bmatrix} 6 & 4 & 2 \\ 5 & 3 & 1 \end{bmatrix} \begin{bmatrix} 18 & 16 & 24 \\ 17 & 15 & 13 \end{bmatrix} f_1 = \begin{bmatrix} 12 & 10 & 8 \\ 11 & 9 & 7 \end{bmatrix} \begin{bmatrix} 24 & 22 & 20 \\ 23 & 21 & 19 \end{bmatrix}$$

(b) Berechnen Sie die Größe der Ausgabe $[o_x, o_y, o_c]$ folgender Convolution:

Shape der Eingabe [20,18,5]
Shape des Filters [3,4,5,1337]
Stride: $s = 2$
Padding: $p = 3$

Lösung:

$$o_x = floor(\frac{(x + 2p_x) - k_x}{stride_x}) + 1 = \frac{20 + 6 - 3}{2} + 1 = 12$$
$$o_y = floor(\frac{(y + 2p_y) - k_y}{stride_y}) + 1 = \frac{18 + 6 - 4}{2} + 1 = 11$$
$$o_c = 1337$$

Aufgabe 3: Rekurrente Netze

Im Folgenden sind die Forward Gleichungen eines sogenannten Peephole-LSTMs gezeigt.

$$f'_t = x_t \cdot W_f + c_{t-1} \cdot W_{Rf} + b_f$$
$$f_t = \sigma(f'_t)$$

$$i'_t = x_t \cdot W_i + c_{t-1} \cdot W_{Ri} + b_i$$
$$i_t = \sigma(i'_t)$$

$$o'_t = x_t \cdot W_o + c_{t-1} \cdot W_{Ro} + b_o$$
$$o_t = \sigma(o'_t)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tanh(x_t \cdot W_c + b_c)$$
$$h_t = \tanh(o_t \cdot c_t)$$

Lösung: $\delta c_t = \delta h_t \cdot (1 - tanh^2(o_t \cdot c_t)) \cdot o_t + \delta c_{t+1} \cdot f_{t+1} + \delta o'_{t+1} \cdot W_{Ro}^T + \delta f'_{t+1} \cdot W_{Rf}^T + \delta i'_{t+1} \cdot W_{Ri}^T$

(a) Berechnen Sie die Gleichung für δc_t

Aufgabe 4: Reverse Mode Automatic Differentation

Gegeben ist die Funktion $L = \cos(x_1 \cdot x_2) + x_1 \cdot x_2 + x_1$

(a) Berechnen Sie den Gradienten der Funktion L händisch

Lösung: $\nabla L = \begin{bmatrix} -\sin(x_1 \cdot x_2) \cdot x_2 + x_2 + 1 \\ -\sin(x_1 \cdot x_2) \cdot x_1 + x_1 \end{bmatrix}$

(b) Berechnen Sie den Gradienten von L an der Stelle (1,2)

Lösung: $\nabla L = \begin{bmatrix} -\sin(1 \cdot 2) \cdot 2 + 2 + 1 \\ -\sin(1 \cdot 2) \cdot 1 + 1 \end{bmatrix} = \begin{bmatrix} 1.19 \\ 0.09 \end{bmatrix}$

(c) Berechnen Sie den Gradienten von L an der Stelle (1,2) mittels Reverse Mode AD

Lösung: Forward:

$$\begin{aligned} v_0 &= x_1 \\ v_1 &= x_2 \\ v_2 &= v_0 \cdot v_1 \\ v_3 &= cos(v_2) \\ v_4 &= v_3 + v_2 + v_0 = L \end{aligned}$$

Backward:

$$\begin{aligned} \frac{\partial L}{\partial v_4} &= 1 \\ \frac{\partial L}{\partial v_3} &= 1 \cdot 1 \\ \frac{\partial L}{\partial v_2} &= -sin(v_2) = -0.91 \cdot 1 + 1 \\ \frac{\partial L}{\partial v_1} &= (-0.91 + 1) \cdot v_0 = 0.09 \cdot 1 = 0.09 \\ \frac{\partial L}{\partial v_0} &= (-0.91 + 1) \cdot v_1 + 1 = 1.19 \end{aligned}$$

Aufgabe 5: Average Pooling

Average Pooling sei definiert, so dass die Operation des Poolings den Average aller Werte nimmt und in den Output schreibt. Beschreiben Sie, wie Sie die Deltas im Backward-Pass ermitteln können.

Lösung:

Für einen Filter mit k Einträgen, muss jedes δy auf k Elemente aufgeteilt werden. Es muss also demzufolge eine Routine existieren, die für einen Output index, den Index des linken oberen Elementes der Filtermatrix ermittelt. Über diesen Index kann man dann die Deltas aufteilen. Da jedes δx von mehreren δy einen Beitrag bekommen kann, ist darauf zu achten immer mit $+=$ zu updaten, und zu Beginn alle Deltas auf 0 zu setzen.