ECE 417 Multimedia Signal Processing
Homework 6

UNIVERSITY OF ILLINOIS
Department of Electrical and Computer Engineering

Assigned: Monday, 11/16/2020; Due: Wednesday, 12/2/2020

Problem 6.1

Suppose you have a recurrent neural network with input $x[n]$, target $y[n]$, output $\hat{y}[n]$, and error metric

$$\mathcal{E} = -\frac{1}{N}\sum_{n=0}^{N-1}\left(y[n]\ln\hat{y}[n] + (1-y[n])\ln(1-\hat{y}[n])\right)$$

where

$$\hat{y}[n] = \sigma(e[n]),$$
$$e[n] = x[n] + \sum_{m=1}^{M-1} w[m]\hat{y}[n-m],$$

and where $\sigma(\cdot)$ is the logistic sigmoid. Write $d\mathcal{E}/dw[3]$ in terms of the signals $y[n]$ and $\hat{y}[m]$. You can invent auxiliary signals such as $\dot{\sigma}[n]$, $\epsilon[n]$, or $\delta[n]$ if you wish, but you need to define them clearly. You may assume that $\hat{y}[n] = 0$ for $n < 0$.

Solution: First step: forward-prop. Assume that $\hat{y}[n]$ has been calculated. Second step: partial derivatives:

$$\epsilon[n] = \frac{\partial\mathcal{E}}{\partial e[n]} = \frac{1}{N}\left(\hat{y}[n] - y[n]\right)$$

Third step: total derivatives:

$$\begin{aligned}\delta[n] &= \frac{d\mathcal{E}}{de[n]}\\ &= \epsilon[n] + \sum_{m=1}^{M-1}\frac{d\mathcal{E}}{de[n+m]}\frac{\partial e[n+m]}{\partial e[n]}\\ &= \epsilon[n] + \sum_{m=1}^{M-1}\delta[n+m]w[m]\dot{\sigma}\left(e[n]\right)\\ &= \epsilon[n] + \sum_{m=1}^{M-1}\delta[n+m]w[m]\hat{y}[n](1-\hat{y}[n])\end{aligned}$$

Fourth step: weight gradient:

$$\begin{aligned}\frac{d\mathcal{E}}{dw[3]} &= \sum_{n=0}^{N-1}\frac{d\mathcal{E}}{de[n]}\frac{\partial e[n]}{\partial w[3]}\\ &= \sum_{n=0}^{N-1}\delta[n]\hat{y}[n-3]\end{aligned}$$

Problem 6.2

Suppose that

$$\begin{aligned}
h_0 &= x^3 \\
h_1 &= \cos(x) + \sin(h_0) \\
\hat{y} &= \frac{1}{2}\left(h_1^2 + h_0^2\right)
\end{aligned}$$

What is $d\hat{y}/dx$? Express your answer as a function of x only, without the variables h_0 or h_1 in your answer.

Solution:

$$\begin{aligned}
\frac{dh_0}{dx} &= 3x^2 \\
\frac{dh_1}{dx} &= \frac{\partial h_1}{dx} + \frac{dh_0}{dx}\frac{\partial h_1}{\partial h_0} \\
&= -\sin(x) + 3x^2\cos(x^3) \\
\frac{d\hat{y}}{dx} &= \frac{dh_0}{dx}\frac{\partial \hat{y}}{\partial h_0} + \frac{dh_1}{dx}\frac{\partial \hat{y}}{\partial h_1} \\
&= h_0\left(3x^2\right) + h_1\left(-\sin(x) + 3x^2\cos(x^3)\right) \\
&= 3x^5 + \left(\cos(x) + \sin(x^3)\right)\left(-\sin(x) + 3x^2\cos(x^3)\right)
\end{aligned}$$

Problem 6.3

Consider a one-gate recurrent neural net, defined as follows:

$$\begin{aligned}
c[n] &= c[n-1] + w_c x[n] + u_c h[n-1] + b_c \\
h[n] &= o[n]c[n] \\
o[n] &= \sigma(w_o x[n] + u_o h[n-1] + b_o)
\end{aligned}$$

where $\sigma(\cdot)$ is the logistic sigmoid, $x[n]$ is the network input, $c[n]$ is the cell, $o[n]$ is the output gate, and and $h[n]$ is the output. Suppose that you've already completed synchronous back-prop, which has given you the following quantity:

$$\epsilon[n] = \frac{\partial E}{\partial h[n]}$$

Find asynchronous back-prop formulas, i.e., find formulas for the following quantities, in terms of one another, and/or in terms of the other quantities defined above:

$$\begin{aligned}
\delta_h[n] &= \frac{dE}{dh[n]} \\
\delta_o[n] &= \frac{dE}{do[n]} \\
\delta_c[n] &= \frac{dE}{dc[n]}
\end{aligned}$$

Solution:

$$\begin{aligned}\delta_h[n] &= \epsilon[n] + \delta_c[n+1]u_c + \delta_o[n+1]\dot{\sigma}(w_o x[n+1] + u_o h[n] + b_1)u_o \\ &= \epsilon[n] + \delta_c[n+1]u_c + \delta_o[n+1]o[n+1](1 - o[n+1])u_o \\ \delta_o[n] &= \delta_h[n]c[n] \\ \delta_c[n] &= \delta_c[n+1] + \delta_h[n+1]u_c\end{aligned}$$

Problem 6.4

Using the CReLU nonlinearity for both σ_h and σ_g in an LSTM, choose weights and biases,

$$\{b_c, u_c, w_c, b_f, u_f, w_f, b_i, u_i, w_i, b_o, u_o, w_o\},$$

that will cause an LSTM to count the number of nonzero inputs, and output the tally only when the input is zero:

$$h[n] = \begin{cases} \sum_{m=0}^{n} \mathbf{1}[x[m] \geq 1] & x[n] = 0 \\ 0 & x[n] \geq 1 \end{cases}$$

where $\mathbf{1}[\cdot]$ is the unit indicator function, and you may assume that $x[n]$ is always an integer.

Solution: First, we only want to generate output when $x[n] \geq 1$, so

$$o[n] = \text{CReLU}(x[n]), \quad \Rightarrow \quad b_o = 0, w_o = 1, u_o = 0$$

Second, we want $c[n]$ to increase by exactly 1, every time $x[n] \geq 1$, i.e.,

$$c[n] = c[n-1] + \text{CReLU}(x[n]),$$

but we know that $c[n]$ is defined to be

$$c[n] = i[n]\text{CReLU}\left(w_c x[n] + u_c h[n-1] + b_c\right) + f[n]c[n-1]$$

so we want $i[n] = 1$ always, $f[n] = 1$ always, and

$$w_c = 1, u_c = 0, b_c = 0$$

Putting it all together, we have

$$\{b_c = 0, u_c = 0, w_c = 1, b_f = 1, u_f = 0, w_f = 0, b_i = 1, u_i = 0, w_i = 0, b_o = 0, u_o = 0, w_o = 1\}$$