CS 577 — Deep Learning — Homework 4

Read these instructions carefully:

- In the LaTeX source code, type your answer in between "`%%% BEGIN ANSWER`" and "`%%% END ANSWER`". For advanced LaTeXusers, you can use your custom macros if you wish by placing them between "`%%% BEGIN MACROS`" and "`%%% END MACROS`" in the header. Do not modify anything else.
- Turn in both your `.tex` file and the generated `.pdf` file.

1 Backpropagation

[5] point(s) — part a:

Answer the question in Section 1.3.7 of `backpropagation.pdf`: How can we calculate $\frac{\partial f}{\partial z_4}(x, y)$ given correct value of $\frac{\partial f}{\partial z_5}(x, y)$?

Answer:
Using the chain rule, we can calculate the derivative of f with respect to z_4 as follows:

$$\frac{\partial f}{\partial z_4} = \frac{\partial f}{\partial z_5}(x, y)\frac{\partial z_5}{\partial z_4}$$

[5] point(s) — part b:

Answer the question in Section 1.3.8 of `backpropagation.pdf`: What is currently stored in `z3.grad` right before `z4._backward()` is called?

Answer:
$\frac{\partial f}{\partial x_3}$ is stored in z3.grad . Using the chain rule, we can calculate the derivative of f with respect to z_3 as follows:

$$\frac{\partial f}{\partial x_3} = \frac{\partial f}{\partial x_8}(x, y)\frac{\partial x_8}{\partial x_3}$$

We know that $\frac{\partial f}{\partial x_8}(x, y)$ is 1, as $z_8 = f(x, y)$ and $\frac{\partial x_8}{\partial x_3}$ is $\frac{\partial (x_7 * x_3)}{\partial z_3}$, so:

$$\frac{\partial f}{\partial x_3} = \frac{\partial (x_7 * x_3)}{\partial z_3}$$

Using the product rule and knowing the derivative of z_3 with respect to z_3 is 1, we get:

$$\frac{\partial f}{\partial x_3} = z_7\frac{\partial z_3}{\partial z_3} + z_3\frac{\partial z_7}{\partial z_3} = z_7 + z_3\frac{\partial z_7}{\partial z_3}$$

The first term is z_7 and the second term is $z_3\frac{\partial z_7}{\partial z_3}$. This last term has not been compute because it depends on the backward pass through z_4. So, the value of `z3.grad` stored before calling `z4._backward()` is z_7. The second term will be added once backpropagation processes earlier nodes.

2 Gradient descent with `ag.Scalar`

[10] point(s) — part a: This is a programming exercise. See `hw4.ipynb`

3 Transformer with `ag.Scalar`

[Bonus 20] point(s) — part a: This is a programming exercise. See `hw4.ipynb`